BEAMFORMING INNOVATIONS FOR WIRELESS COMMUNICATION AND OPTICAL COMPUTATION

YANG XUEYUAN

PhD

The Hong Kong Polytechnic University

2025

The Hong Kong Polytechnic University
Department of Computing

Beamforming Innovations for Wireless Communication and Optical Computation

Yang Xueyuan

A thesis submitted in partial fulfillment of the requirements for
the degree of Doctor of Philosophy
August 2024

CERTIFICATE OF ORIGINALITY

I hereby declare that this thesis is my own work and that, to the best of my knowledge and belief, it reproduces no material previously published or written, nor material that has been accepted for the award of any other degree or diploma, except where due acknowledgment has been made in the text.

Signature: _____

Name of Student: Yang Xueyuan

Abstract

Beamforming, originally developed for radar applications during the mid-20th century, has evolved significantly to become a cornerstone in modern communication and signal processing technologies. This advanced technique enhances signal directionality and focus, which is crucial for minimizing interference and maximizing the efficiency of transmission systems. This dissertation investigates the extensive applications and transformative potential of beamforming in contemporary technological fields. This investigation particularly focuses on its integration into wireless communication systems and optical computing.

The first major application discussed is Transfer Beamforming (TBF) for wireless communication systems, where it revolutionizes the management of signal energy, enhancing data transmission range and system reliability in densely populated tag environments like warehouses. By leveraging semi-active tags as initial sniffing tools, TBF facilitates the beamforming process to transfer energy effectively to passive tags, substantially increasing their reading range and reducing miss reading rates. Prototype testing in a crowded warehouse achieved a 99.9% inventory coverage rate, demonstrating a power transmission improvement of 6.9 dB and a doubling of inventory speed compared to existing methods.

Additionally, the dissertation extends the application of beamforming principles to the realm of optical computing, specifically through the development of Binary Optical Neural Networks (BONNs). By leveraging the beamforming characteristic, BONNs

selectively illuminate specific regions when processing different data inputs, thereby effectively accomplishing classification tasks. BONNs employ binarized weights for low-cost fabrication and are capable of processing large-scale data with significantly lower energy requirements. Prototypes show that BONNs consume up to 2,405 times less energy than conventional Electronic Neural Networks(ENNs) while maintaining an average recognition accuracy of 74% across various datasets. The reduction in layer manufacturing costs to just 0.13 per layer presents a scalable, cost-effective alternative to traditional computational models.

This dissertation demonstrates the impact of beamforming innovations on both wireless communication systems and optical computation, reflecting a significant evolution of beamforming. Through the implementation of Transfer Beamforming (TBF) in dense wireless environments and the development of Binary Optical Neural Networks (BONNs), this research highlights the adaptability and efficiency of beamforming techniques in modern technological applications. TBF's ability to enhance data transmission accuracy and the potential of BONNs in processing large datasets with minimal energy displays beamforming's potential to revolutionize communication and computational paradigms. These advancements not only improve system efficiencies but also pave the way for future applications, solidifying beamforming's role as a cornerstone technology that bridges the gap between traditional signal processing and next-generation network and computational technologies.

Publications Arising from the Thesis

1. Xueyuan Yang, Zhenlin An, Qingrui Pan, Lei Yang, Yulong Fan, and Dangyuan Lei, "Binary Optical Machine Learning: Million-Scale Physical Neural Networks with Nano Neurons", in *Proc. of ACM MobiCom*(2024).

2. Xueyuan Yang, Zhenlin An, Xiaopeng Zhao, and Lei Yang, "Transfer Beamforming via Beamforming for Transfer", in *IEEE Transactions on Mobile Computing(TMC)* (2023).

3. Xueyuan Yang, Zhenlin An, Xiaopeng Zhao, and Lei Yang, "Transfer Beamforming via Beamforming for Transfer", in *in Proc. of IEEE INFOCOM* (2023).

4. Qingrui Pan, Zhenlin An, Xueyuan Yang, Xiaopeng Zhao, and Lei Yang, "RF-DNA: large-scale physical-layer identifications of RFIDs via dual natural attributes", in *Proc. of ACM MobiCom*(2022)

Acknowledgments

First and foremost, I extend my deepest gratitude to my supervisor, Dr. Lei Yang, whose invaluable guidance was essential in shaping the direction and execution of this thesis. His profound knowledge and meticulous attention to detail have not only been crucial in my development as a scholar but have also faithfully guided me through the most challenging aspects of my research. His steady devotion and personal commitment to academic excellence were as crucial as his scholarly guidance, constantly inspiring me to persevere even when the path seemed most daunting.

I am deeply thankful to my colleagues in the research group, including the postdoc and PhD students. Special thanks go to my coauthors, Dr. Zhenlin An, Ms. Qingrui Pan, and Mr. Xiaopeng Zhao, whose companionship and insightful peer reviews were invaluable. Their diverse perspectives and constructive critiques have significantly broadened my understanding of my research topics and enriched my academic experience. I'm also grateful to Mr. Jingyu Tong, Mr. Sicong Liao, Mr. Zheng Gong, Mr. Zhimin Mei, Mr. Fengrui Zhang, Mr. Zhicheng Wang, Mr. Shen Wang and all collaborators for their selfless help and emotional support throughout my PhD journey.

I would like to extend my sincere thanks to my mother, Ms. Xingying Yan, and my father, Mr. Fengxiang Yang. whose love and support have been the foundation of everything I am. Thank you both for your endless sacrifices. I cannot express enough gratitude to my friends, Ms. Jiareng Chen, Ms. Ningning Hou, and Ms. Kaiyan Cui

for their firm support throughout my PhD journey. Their selfless help and constant encouragement have been my anchor during times of stress and uncertainty.

I would also like to take a moment to acknowledge my own efforts and persistence. Completing a PhD requires not just intellectual effort but a deep personal commitment and resilience. Thanks to myself for the countless hours of study and the perseverance through setbacks. This journey has transformed me, and I am profoundly grateful for the insights and maturity I have gained along the way.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

Beamforming, a technique that manipulates the phase and amplitude of an antenna array to direct energy toward specific receivers, has revolutionized both wireless communication and optical computation. This advanced signal processing method, which originated in radar systems, has adapted over decades to meet the increasing complexities of modern technological demands. Today, it plays a crucial role in optimizing the performance of various communication and computational systems. As shown in Fig. 1.1, the thesis focuses on beamforming applicaiton in wireless communication and optical computation.

In the realm of wireless communications, particularly within RFID systems, the importance of beamforming cannot be overstated. By focusing transmitted energy in precise directions, beamforming significantly enhances the communication range and accuracy of RFID systems[27, 18, 69, 55, 116, 119, 111]. Beamforming in wireless communication can help constructive interference in desired region to avoid blind spots and activate tags more easily. This capability is critical in environments such as large warehouses and retail spaces, where efficient and accurate tag detection is essential for inventory management and asset tracking. Beamforming helps overcome common challenges like signal interference and fading, ensuring that even tags lo-

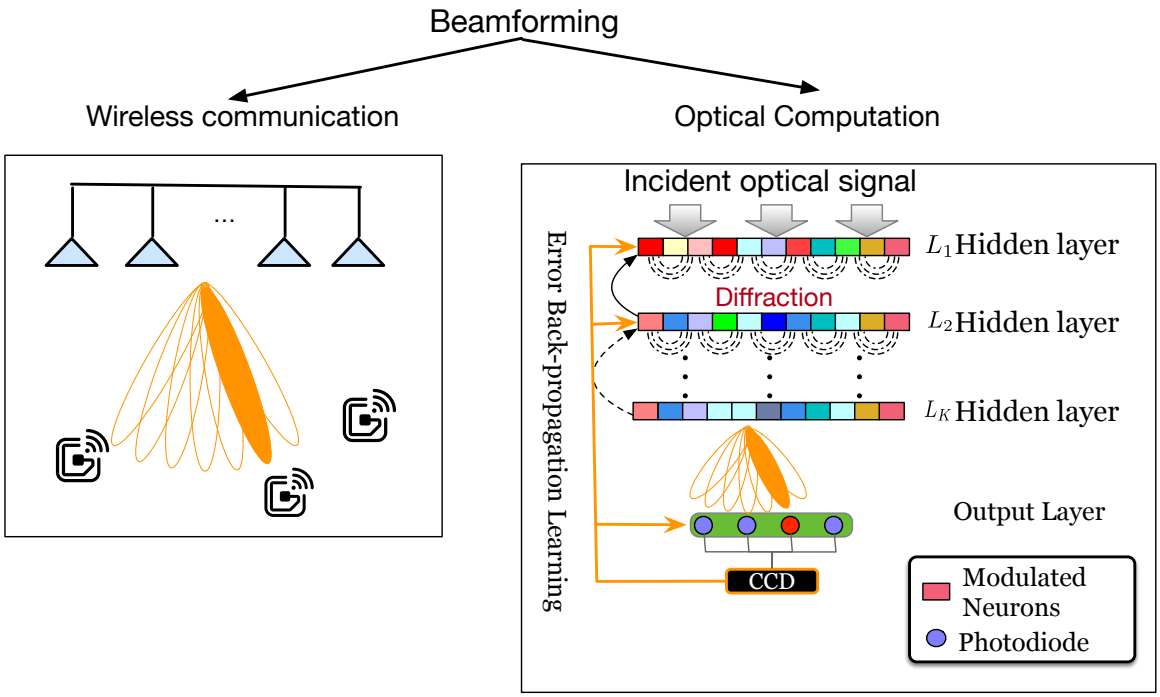


Figure 1.1: Beamforming in wireless communication and optical computation

cated at a long distance or in densely packed settings are reliably detected. This technology has been widely applied to various scenarios, including real-time location systems in healthcare for tracking equipment and patient monitoring, as well as in logistics for streamlining supply chain operations, demonstrating its wide applicability in enhancing connectivity and data accuracy across diverse sectors.

The application of beamforming extends into the field of optical computing, where it is used to manipulate light waves to process data at the speed of light. Here, beamforming technologies enable the implementation of Optical Neural Networks (ONNs), which perform complex calculations with high throughput and low latency. ONNs can beam the energy to the predefined region to preform classfication and saliency detection task with beamforming. As there are millions of neurons in a hidden layer, error back-propogation has been applied to find optimal solution like the eletronic neural network. Beamforming allows ONNs to perform complex computational tasks more

efficiently than traditional electronic computing methods, opening new possibilities in data processing and artificial intelligence with optical signals.

In summary, beamforming stands as a transformative technology that significantly enhances both wireless communication and optical computation. Its ability to direct and manage signals and light with unprecedented precision brings considerable improvements in system performance, energy efficiency, and operational scope. As we continue to explore and expand its applications, beamforming is set to remain at the forefront of technological advancements, shaping the future of digital communication and computation.

1.1 Beamforming in Wireless Communication

In this section, we will explore the concept of beamforming within wireless communication, focusing on its evolution and its specialized applications, particularly in RFID systems. We will trace the development of beamforming from its early implementations to its current status as an integral technology for improving the performance of wireless communication. This discussion will highlight how beamforming enhances signal clarity and extends the operational range, significantly impacting the effectiveness and reliability of wireless systems across various industries.

1.1.1 Evolution of Beamforming

The evolution of beamforming in wireless communication represents a significant technological advancement that has profoundly impacted the efficiency and functionality of modern communication systems. Beamforming improves the quality of communication and increases the communication range, making it an invaluable tool in both congested wireless environments and in applications requiring precise directional sensitivity. Initially developed for military radar applications, beamforming

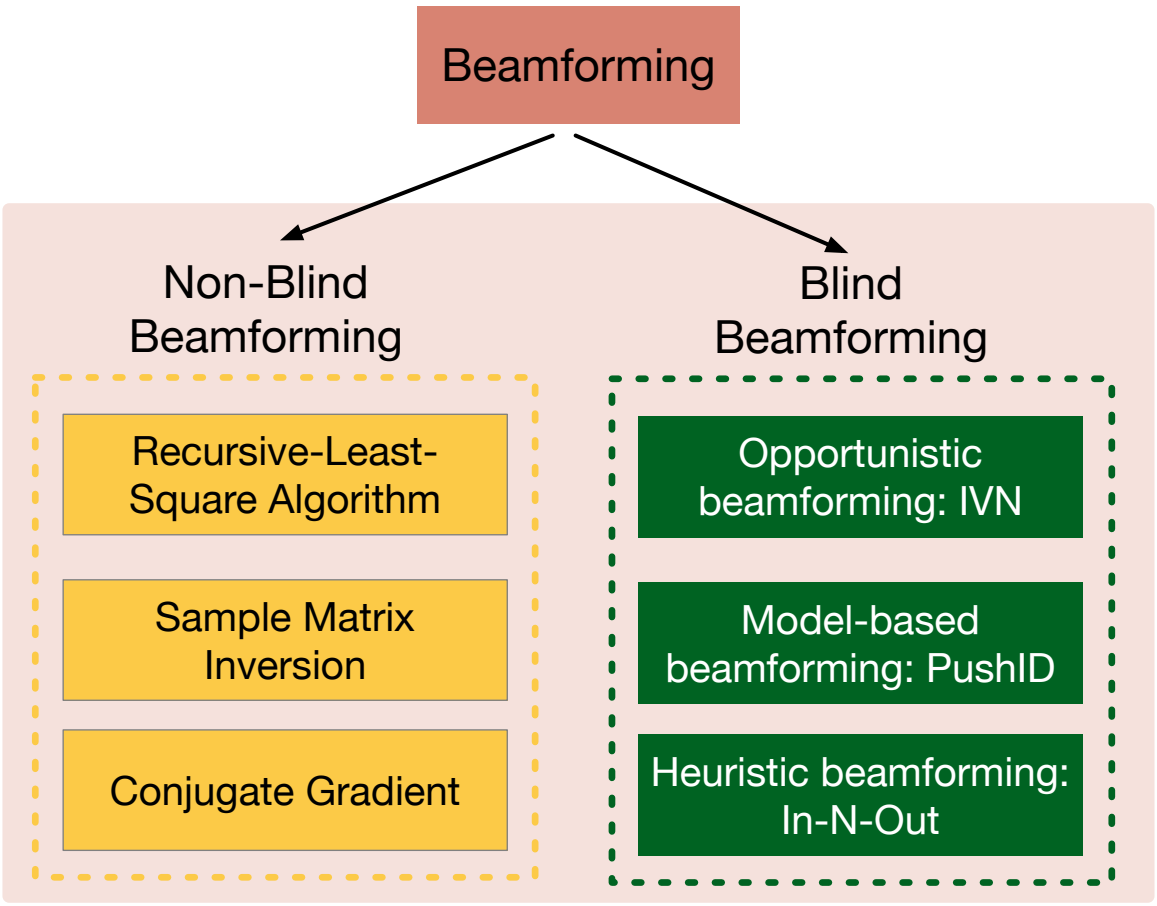


Figure 1.2: Beamforming classification in wireless communication

has undergone substantial transformations to become a fundamental component in the architecture of contemporary wireless communication. This section explores the historical progression of beamforming technology, highlighting its transition from a hollow application to a critical element in achieving high-performance wireless communications.

As shown in Fig. 1.2, beamforming technology is primarily divided into two major categories: non-blind algorithms and blind algorithms. The former, non-blind adaptive algorithms, make use of a reference signal to monitor the performance of the beamforming weights. Once these weights are set, the response from this configuration is

assessed against the reference signal. The discrepancy between the two, measured as an error difference, serves as a metric to evaluate performance. Through multiple iterations, this error difference is progressively reduced until the beam accurately targets the intended direction. Some of the typical non-blind beamforming algorithms include the Recursive-Least-Square (RLS) algorithm [18], the Sample Matrix Inversion [69], and the Conjugate Gradient [55].

Despite the precision that non-blind beamforming can achieve, it faces significant challenges that limit its widespread adoption. One significant limitation is its dependency on prior knowledge of channel characteristics or the directions of arrival (DOAs) of incoming signals. In real-world settings, securing accurate and timely channel information can be problematic, especially in dynamic environments where the channel conditions may fluctuate rapidly. This requirement not only complicates the beamforming process but also adds complexity to the system's overall operation due to the necessity to handle and process the reference signals continually.

Furthermore, non-blind beamforming is inherently more suited to static and predictable environments. In scenarios characterized by frequent or substantial changes, the performance of non-blind beamforming algorithms can be significantly damaged. The algorithms' effectiveness diminishes as they struggle to adapt to the new conditions, leading to suboptimal beamforming outcomes. This limitation makes non-blind beamforming less ideal for applications in highly variable environments where adaptability is key to maintaining consistent performance.

Despite these challenges associated with non-blind beamforming, the field of beamforming also includes alternative strategies that operate without the dependency on pre-acquired channel information. These are known as blind beamforming algorithms, which are particularly advantageous in dynamic or uncertain environments where obtaining accurate channel data is not feasible. Blind beamforming offers a distinct approach, focusing on real-time signal adaptation without the need for prior channel knowledge.

Blind beamforming algorithms stand out for their ability to operate without predetermined channel state information. These algorithms are crucial in scenarios where acquiring such data is impractical or impossible. Broadly categorized into three types: opportunistic beamforming, model-based beamforming, and heuristic beamforming. Each type of blind beamforming uses distinct mechanisms to adapt to the ever-changing wireless channel conditions.

Opportunistic beamforming techniques, such as IVN [74], dynamically manipulate the carrier signal to suit complex and variable channel conditions. It varies each antenna's output frequency, which creates time-varying envelopes at the receiver. These variations facilitate constructive interference, effectively reducing or eliminating coverage blind spots, and it's a critical improvement in maintaining continuous and reliable communication.

Model-based beamforming strategies take a different approach by constructing sophisticated models of the environment. These models are built through the analysis of signals collected directly from the environment or by using auxiliary information such as data from other channels or frequency bands. For example, PushID [116] exploits channel reciprocity, modeling the spatial distribution of signal power to optimize the number of beamforming vectors, reducing overlap and enhancing signal clarity. SpaceBeam [119] develops a spatial model that identifies and utilizes invariant structures specific to mmWave channels, enabling rapid beam selection through precomputed lookup tables designed for lidar-integrated mmWave communications. Another innovative approach detailed in research by Vasisht [112] infers channel behavior in one frequency band based on the observed data in another, leveraging the stability of physical signal paths across different frequencies to enhance predictive accuracy.

However, both opportunistic and model-based strategies can fall short in ensuring optimal power transmission in complex environments. To address this, heuristic beamforming methods iteratively gather feedback from receivers to refine and optimize

beamforming strategies such as In-N-Out [38]. Although effective, this feedback-driven iterative process can be time-intensive and inefficient, particularly unsuitable for large-scale or rapidly changing scenarios.

Despite the significant advantages offered by blind beamforming, its efficiency is occasionally challenged in environments characterized by high noise levels, significant signal variability, or extreme dynamism. The continuous advancement of algorithmic strategies and computational technologies is progressively mitigating these challenges, thereby enhancing the adaptability, efficiency, and overall utility of blind beamforming techniques across various application scenarios. This continual advancement not only expands the practical applications of blind beamforming but also strengthens its position as a key technology in modern wireless communication systems.

1.1.2 Beamforming Application in Wireless Systems

RFID beamforming represents an advancement in signal processing technology that substantially enhances the utility and application of RFID systems across a broad spectrum of industries. By focusing the signal in a specific direction, actively suppressing interfering signals, and significantly improving the signal-to-noise ratio, RFID beamforming promotes the performance of communication systems across domains such as antenna, radar, and sonar.

The primary technical advantage of RFID beamforming lies in its ability to direct the transmission energy toward specific receivers while minimizing energy in unwanted directions. This capability not only extends the operational range of RFID systems but also ensures that signal integrity is maintained even in environments full of noise and interference. Such precision is crucial in densely populated IoT environments, where the coexistence of multiple wireless devices could potentially lead to signal interference and cross-talk, undermining the reliability and efficiency of communication networks.

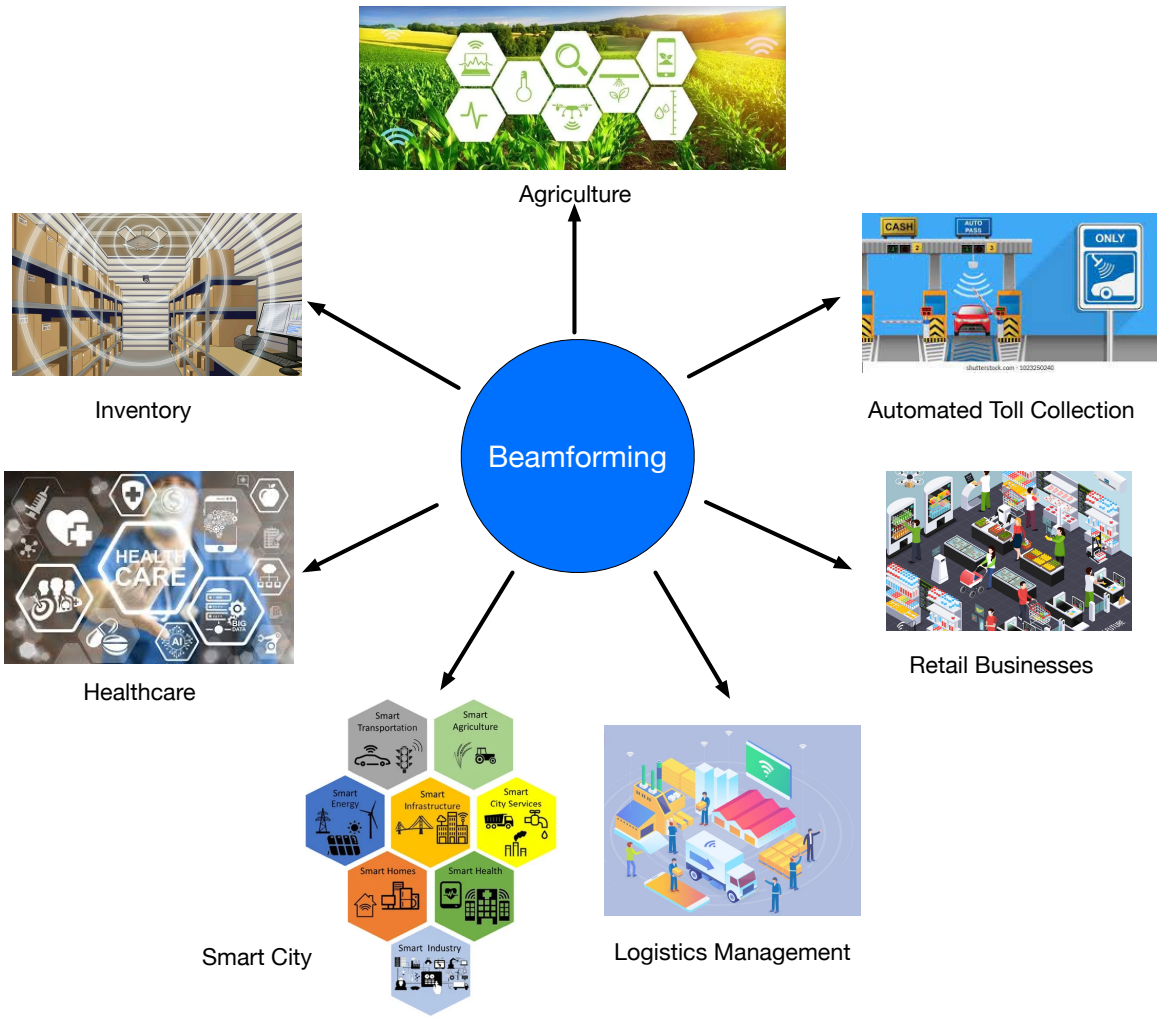


Figure 1.3: The application scenarios of Beamforming

The versatility of beamforming technology has led to its widespread adoption across various sectors, each benefiting from improved communication reliability and system efficiency. As shown in Fig. 1.3, beamforming technology has been widely applied in many scenarios, such as healthcare, smart agriculture, etc.

For example, RFID beamforming is crucial in logistics and supply chain management [94], where the accurate and efficient tracking of goods is essential. It allows for extended range and precision in tracking items as they move through various points

in the supply chain, from warehouses to distribution centers and retail outlets. This enhanced tracking capability helps reduce losses, improve inventory accuracy, and streamline operations.

In healthcare facilities, RFID beamforming can be used to monitor the location and status of critical equipment, medications, and even staff and patients [85]. The precision and reliability provided by beamforming ensure that medical assets can be tracked in real time, improving resource allocation and patient care while minimizing the loss and misplacement of expensive medical devices.

Beamforming technology allows retail businesses to enhance customer experiences through better inventory management and item tracking [22]. Retailers can use RFID systems to ensure product availability, reduce theft, and even implement advanced marketing strategies like smart shelves that detect when customers pick up items and provide relevant information or promotions.

Beamforming is critical in industrial settings for tracking equipment and inventory across extensive facilities [98], including challenging environments that may impede standard RFID systems. Its robustness and reliability support asset management in industries such as manufacturing and construction, ensuring that operations run smoothly and without unnecessary downtime.

RFID beamforming contributes to the development of smart city infrastructure by enabling the efficient management of city assets such as public transportation vehicles, municipal equipment, and emergency services [106]. Enhanced tracking capabilities facilitate better urban planning, improved public safety, and more efficient use of resources.

By increasing the accuracy and reliability of vehicle identification at toll booths [52], RFID beamforming enables seamless, high-speed toll collection, thereby reducing traffic congestion and enhancing the throughput of toll roads.

In the agriculture sector, RFID beamforming can be used to track livestock and equip-

ment over large areas[93]. It enhances the management of farm assets, improves the monitoring of animal health, and assists in the optimization of agricultural operations through better resource allocation.

The integration of RFID beamforming into these sectors not only highlights its importance but also underscores its potential to drive significant improvements in operational efficiency, security, and data accuracy. By making use of the strengths of beamforming, industries are better equipped to meet the challenges of an interconnected IoT world, paving the way for more innovative and effective solutions in wireless communication and asset management.

1.2 Beamforming in Optical Computing

In this section, we investigate the application of beamforming technologies in the realm of optics, specifically a field we denote as optical neural networks. We begin by exploring the fundamental connection between traditional beamforming methods and their adaptation to optical systems, illuminating how these principles are harnessed to enhance data processing and signal clarity in optical neural networks.

Following this, we shift our focus to the broader development of optical neural networks. We will explore the influential advancements that currently define this field, emphasizing both the latest technological breakthroughs and theoretical contributions driving innovation. Our discussion will detail the newest materials and design strategies that enhance signal processing alongside the integration of sophisticated algorithms that improve network efficiency. This overview aims to provide a clear understanding of how optical neural networks are evolving to address increasingly complex computational challenges.

In thefollowing, We explore the applications of optical neural networks, focusing on their use in image recognition, data transmission, and medical diagnostics. These

networks provide speed and accuracy improvements in processing visual information, enhancing bandwidth in telecommunications, and supporting early disease detection through advanced image analysis.

We also highlight persistent challenges that researchers and developers face and recent innovations that have pushed the field forward, including breakthroughs in materials science that enable more efficient light manipulation, and advancements in algorithm design that improve network performance. By providing both a historical context and a forward-looking perspective, we aim to give a thorough overview of the optical network landscape and its impact on the future of optical computing.

1.2.1 Optical Neural Network: Beamforming Application in Optical Computing

We adapt the beamforming technique for use in the light spectrum, leading to innovative applications such as optical neural networks for many tasks. The whole working process of the optical beamforming is illustrated in Fig. 1.4. In this setup, the coherent parallel light is modulated by the input layer. The incident light is forwarded to multiple modulation layers to modulate the amplitude or phase of the input signal, which is modulated by optical devices such as spatial light modulators (SLMs) or other types of metasurfaces [68]. The nodes on the optical devices function as antennas that change the characteristics of signals to perform beamforming. This modulation directs the signal in a specific direction, illuminating targeted regions on imaging devices like CCDs or CMOS sensors, which function as the output layer.

As illustrated in the left figure in Fig. 1.5, the system defines ten regions corresponding to ten different classification categories for the output CCD images. The region that receives the maximum energy determines the classification outcome. In this example, the second region exhibits the highest energy and it indicates that the input belongs to class 1. The objective of the recognition is to maximize energy in the designated

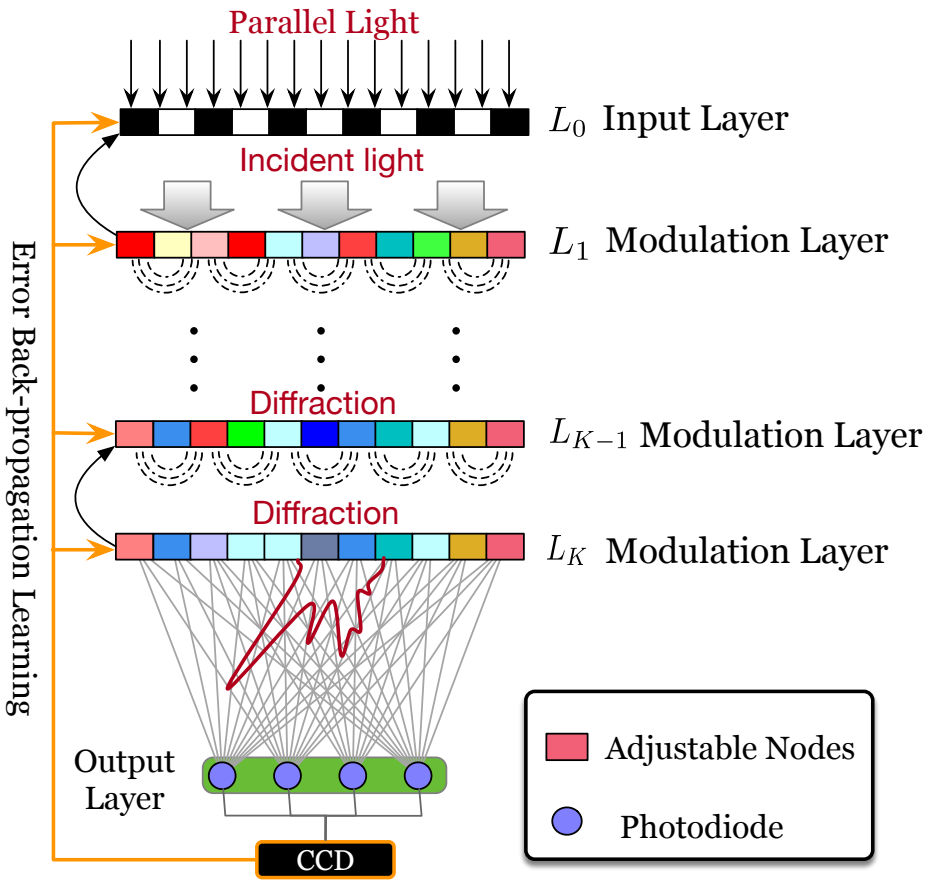


Figure 1.4: Working process of Optical Beamforming.

target region while minimizing it in others by adjusting the parameters of the nodes, a principle akin to traditional beamforming.

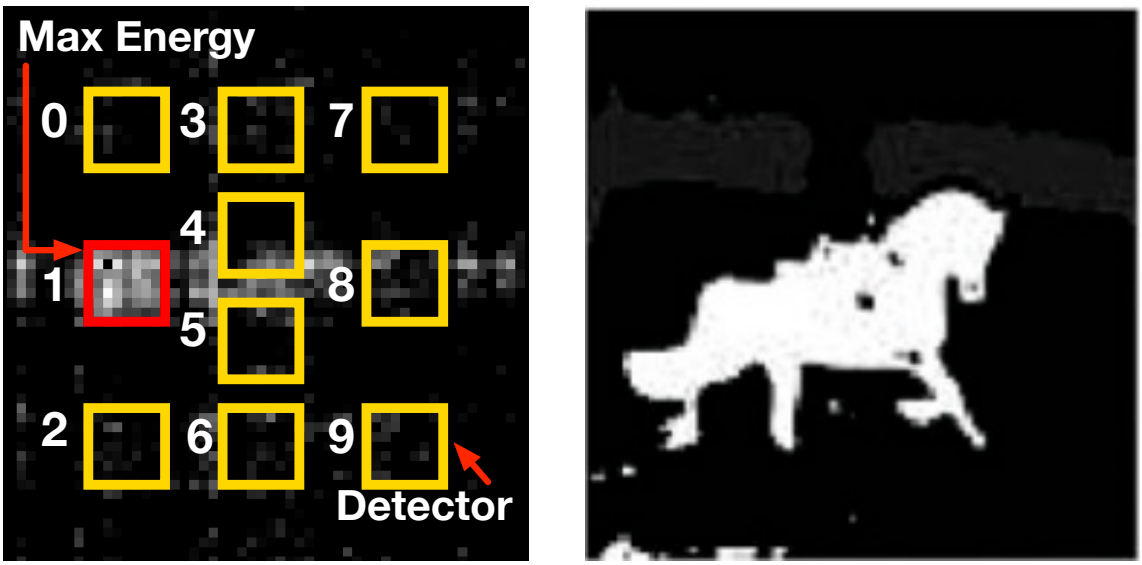
Another application of this technique is saliency detection, depicted in Fig. 1.5. The task focuses on directing energy to a specific region of interest while weakening others, making the target region extremely visible. This approach enhances both recognition tasks and saliency detection by utilizing the principles of beamforming.

While optical beamforming shares similarities with its counterpart in wireless beamforming systems, it requires at least hundreds of adjustable array elements to effectively perform in optical computing. Calculating an optimal solution for these arrays

is complex. However, inspired by traditional electronic neural networks, which employ extensive computations across numerous neurons, the optical beamforming array elements can also be optimized using the error back-propagation algorithm. Given these parallels, we describe beamforming application in optical computing as an "Optical Neural Network," wherein each array element plays a critical role analogous to a neuron in electronic networks.

Compared with electronic neural networks, optical neural networks(ONNs) have several unique advantages. Initially, ONNs operate at the speed of light, using photons rather than electrons for signal transmission. This fundamental difference enables ONNs to achieve remarkably fast data processing speeds. Moreover, ONNs inherently support parallel processing as multiple light beams can travel through the same medium simultaneously without interference. This ability allows for simultaneous computations, significantly enhancing throughput and computational efficiency. Subsequently, Optical systems are typically more energy-efficient than their electronic counterparts. They operate with minimal energy loss since photons travel through optical media like fibers or free space without the need for a lot of power, which is often consumed in overcoming electrical resistance in electronic systems. Additionally, ONNs generate less heat due to reduced energy loss, which diminishes the need for extensive cooling infrastructure. This not only saves additional energy but also simplifies the system design. Finally, ONNs can handle higher bandwidths than electronic systems, thanks to the high frequency of light waves. This allows more data to be processed simultaneously, making optical systems particularly effective for high-throughput applications. Furthermore, optical components can be miniaturized and integrated into dense configurations more readily than electronic circuits. This advantage is crucial for developing compact devices with high integration levels, benefiting applications where space and power efficiency are critical.

In summary, optical neural networks stand at the forefront of technological innovation, offering significant advancements in speed, efficiency, and scalability over traditional



(a) Recontion Task (b) Saliency Detection

Figure 1.5: Beamforming application in Optical Computing. (a) is for the recognition task, and (b) is for the saliency detection.

electronic systems by utilizing the beamforming principle. As we continue to explore and develop these networks, their potential to transform a wide range of applications is undeniable, marking them as a crucial area of study in the evolution of computational technologies.

1.2.2 Development of Optical Neural Networks

Following over fifty years of research and development, there now exists a wide range of theoretical models [95, 96, 62] for Artificial Neural Networks (ANNs). These models have been extremely successful in numerous fields, such as object recognition [62], object tracking [83], image generation [44], and natural-language processing [108]. Currently, most ANNs employed in practical applications are software simulations running on traditional electronic von Neumann architecture computers. Despite sig-

nificant advancements made with this approach [60, 100, 115], it faces inherent challenges: notably, as the number of transistors on central processing units (CPUs) and graphical processing units (GPUs) grows exponentially, power consumption poses a significant challenge in training models. The end of Dennard scaling [34] implies that reducing transistor size no longer leads to lower power consumption, meaning even minor enhancements in CPU or GPU performance can result in significant increases in energy usage and heat production. For example, to simulate an ANN on the scale of the human brain in real-time within the von Neumann architecture would require at least 500 MW of power [76] and utilize massive supercomputers. These constraints highlight that the issue of efficiently training large, data-intensive ANNs to perform specific tasks on von Neumann systems consuming substantial time and energy is unlikely to improve significantly in the foreseeable future.

As the quest for more efficient computing paradigms continues, optical neural networks emerge as another transformative technology with the potential to further enhance the capabilities of Artificial Neural Networks (ANNs) by fusing optical computing and beamforming. Optical neural networks, which use photons rather than electrons to perform computations based on the natural physical law, offer distinct advantages that could address some of the current limitations of traditional and emerging computing technologies.

Unlike electronic systems, Optical neural networks utilize light to process and transmit information, which can significantly reduce energy consumption and heat generation. This approach allows for extremely high-speed data processing and minimal latency due to the speed of light, far surpassing that of electrical signals. Optical components such as lasers, lenses, and mirrors are used to perform operations through various properties of light such as interference and diffraction, enabling parallel processing and fast data handling capabilities. Optical neural networks can efficiently perform matrix multiplication and convolution operations much faster and with less energy than electronic counterparts which are the core components of deep learning

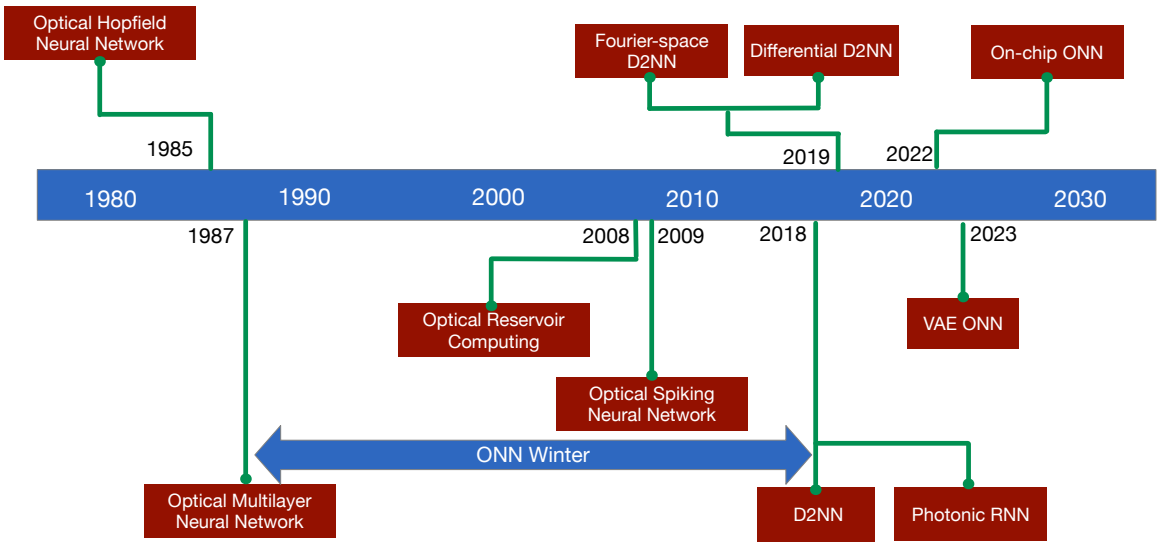


Figure 1.6: The Development of Optical Neural Network(ONN)

algorithms. This efficiency originates from the capability to execute operations simultaneously and at the speed of light, which is vital for training larger and more complex neural network models that require intensive data processing and substantial computational resources.

An Optical Neural Network (ONN) utilizes optical computing to emulate the framework of an Artificial Neural Network (ANN), presenting an innovative approach that merges the strengths of both paradigms while overcoming the constraints imposed by Moore's Law. As depicted in Fig. 1.6, the initial model of an ONN, known as the Optical Hopfield neural network, was introduced in 1985 by Psaltis et al [86]. In this pioneering model, an array of light-emitting diodes (LEDs) is used to represent vectors, while a spatial light modulator (SLM) acts as the matrix. Two cylindrical lenses adjust the direction of the light's travel, and a photodetector array, positioned perpendicular to the LED array, performs the summation operation by measuring light intensity. The design of the optical vector-matrix multiplier, with input nodes aligned linearly, however, does not fully capitalize on the inherent parallel processing

capabilities of optical computing.

The concept of an optical Multilayer Neural Network was initially introduced in 1987 by Masatoshi et al. [40]. This pioneering work utilized multiple Spatial Light Modulators (SLMs) to construct an optical system capable of associative memory functions and error backpropagation, featuring adaptable learning capabilities. Despite this early innovation, significant advancements in the field of optical neural networks (ONNs) stagnated for a considerable period, a phase referred to as the "ONN Winter." During this time, exceptions included developments in Optical Reservoir Computing and Optical Spiking Neural Networks.

Optical Reservoir Computing emerged in 2008 when Vandoorne et al. [111] designed a system integrating coupled Semiconductor Optical Amplifiers, which introduced the hyperbolic tangent (tanh) function into photonics for the first time, targeting pattern recognition tasks. The following year, David Rosenbluth and his team [95] pioneered the first all-optical fiber spiking neural network. This network was specifically designed around the physical properties of semiconductor devices, effectively merging the benefits of both analog and digital computational elements to implement an integrated-and-fire neuron model.

After decades, Lin et al. [68] introduced a passive all-optical diffraction neural network (D2NN) operating at Terahertz wavelengths, designed for image recognition tasks. This innovative method marked a significant breakthrough due to its minimal energy consumption originating from the use of passive devices and its cost-effective manufacturing process. The architecture of this system featured multiple cascading mask layers, closely resembling the multi-layered structure of artificial neural networks. In this setup, the 'neurons' were modulated by altering the thickness and transmissivity of phase masks, showcasing a novel approach that opens up fresh perspectives on the capabilities and future of optical neural networks.

In the same year as previous innovations, a significant advancement was made with

the proposal of the Photonic Recurrent Neural Network (Photonic RNN) [23]. This model utilized micro-mirror devices and Spatial Light Modulators (SLMs) with passive weights, characterized by its high energy efficiency, making it particularly suitable for time-series prediction tasks.

Addressing a fundamental challenge in optical neural networks (ONNs), the differential D2NN [64] was introduced to overcome the issue of non-negativity caused by photodetectors that are only sensitive to light intensity and cannot detect negative values. This network cleverly utilizes both negative and positive detectors, comparing their signals to accurately achieve classification results, thus enhancing the ONN's ability to handle complex computational tasks that require differentiation between various signal types.

Further expanding the capabilities of D2NNs, the Fourier-space D2NN [121] was developed to process tasks involving salient-object detection, which are challenging to perform in real space. This system employs a "4f system", consisting of two convex lenses and a photorefractive crystal (SBN:60), recommended for its nonlinear activation properties. This setup not only enhances the system's ability to focus on important visual features but also significantly boosts processing efficiency through optical computations.

The introduction of the On-Chip ONN [19] marked a critical development in the field, establishing the first end-to-end on-chip photonic neural network. This groundbreaking technology ensures uniform distribution of supply light across all neurons within the network, which standardizes the optical output range across various layers. Such uniformity is vital for scaling the network to accommodate a larger number of layers, potentially revolutionizing how optical neural networks are designed and implemented, paving the way for more compact, efficient, and scalable ONN architectures.

The VAE ONN [30] is an innovative design centered around a high-throughput pro-

cessing system that operates using an all-optical variational autoencoder. This model effectively transforms the process of image transmission into a functioning optical generative neural network. The architecture of this system is crafted to optically encode incoming information into a compressed and encrypted optical latent space, facilitating secure transmission. Furthermore, it can optically decode the distorted signals received from this encrypted domain, reconstructing them into images. Notably, this setup significantly reduces computational latency by over four orders of magnitude compared to current leading-edge devices, enhancing efficiency dramatically.

In conclusion, the advancements in optical neural network (ONN) technologies, as demonstrated by the development of various innovative systems such as Photonic RNNs, differential D2NNs, Fourier-space D2NNs, On-Chip ONNs, and the groundbreaking VAE ONN, highlight a significant evolution in the field of optical computing. These technologies not only push the boundaries of what is possible in terms of processing speed and energy efficiency but also pave the way for new applications across various domains requiring complex computational capabilities. Each of these developments addresses specific challenges inherent to traditional and optical computing methods, proposing novel solutions that enhance performance, reduce power consumption, and expand the potential for future scalability. As the field continues to advance, it is anticipated that optical neural networks will play a vital role in the next generation of computational technology, transforming both theoretical research and practical applications. The ongoing innovation in ONN design and functionality is poised to overcome current limitations and set new benchmarks for what these powerful systems can achieve.

1.2.3 Significance of Optical Neural Networks in application

Optical Neural Networks (ONNs) represent a significant shift from traditional neural network architectures by leveraging light for computation and data transmission.

ONNs offer several advantages over electronic systems, including higher processing speeds, lower latency, high bandwidth, and reduced power consumption. So ONNs are very promising for applications.

ONNs are particularly effective in applications requiring the rapid manipulation and analysis of visual data. Their capability to process information at the speed of light is essential for real-time image recognition, object detection, and computer vision tasks. These are crucial in a variety of settings, including autonomous vehicles, where split-second decision-making can be life-saving, and in surveillance systems, where rapid response to dynamic situations is necessary. Additionally, ONNs are utilized in augmented reality technologies to improve interaction with real-world and digital objects in real-time.

In the field of telecommunications, ONNs can enhance the performance of optical fiber networks. They manage data directly in its optical form, which significantly reduces latency and increases bandwidth. This direct handling avoids the need for conversion between optical and electronic signals, streamlining the data transmission process and enhancing throughput, which is vital for modern communication networks that demand high-speed data flow.

Financial institutions could leverage ONNs for executing high-speed trading algorithms and complex risk assessment models. The ability to analyze market data at high speeds allows for real-time responses to market changes, essential in high-stakes trading environments. Moreover, ONNs are adept at complex signal processing tasks such as noise reduction and signal enhancement. This capability is invaluable in various engineering applications, improving the clarity and quality of audio and video transmissions, which is vital in media, communications, and safety systems.

ONNs can also accelerate computational-intensive tasks such as scientific simulations of weather patterns or astrophysical phenomena. These simulations often require the handling of vast amounts of data at high speeds, making ONNs ideal for such

purposes.

In conclusion, optical neural networks (ONNs) could revolutionize various industries with their unparalleled speed and efficiency. Applied across fields such as telecommunications and financial modeling, ONNs enhance performance and enable new functionalities where traditional systems are limited. As these technologies continue to advance, they promise to transform many sectors, driving innovation and addressing complex challenges effectively.

1.2.4 Current Challenges and Innovations in Optical Neural Networks

Applying Optical Neural Networks (ONNs) to vast industry scenario presents a range of technical challenges that must be addressed to realize their full potential. One of the primary hurdles is the compatibility of ONN systems with current electronic and digital technologies.

The development and deployment of ONNs require advanced optical components that are not typically used in standard electronic circuits. For example, all-optical neural network(AONN) [131] created the input layer and hiddeen with multiple spatial light modulators (SLMs) and lens. They necessitates precise engineering and alignment to function effectively. SLM is quite expensive and the size of SLM is too large to be integrated into existing electronic systems.

The integration of Optical Neural Networks (ONNs) into existing information systems encounters significant challenges, primarily due to the absence of standardized protocols for optical data transmission and processing. Unlike conventional systems that operate based on well-established electronic signal standards, ONNs necessitate intricate signal conversion processes. For instance, [124] describe a serial electro-optical neural network (TS-NN), which is a hybrid photoelectric fully connected neural net-

work undergoing multiple photoelectric conversions, thereby introducing potential errors during data processing. Consequently, the urgent development or modification of existing standards is required to accommodate the unique properties of optical signals and ensure their seamless integration and operational reliability.

Furthermore, the fabrication of neuron layers for ONNs presents a significant financial barrier to broader adoption. Spatial Light Modulators (SLMs) and metasurfaces are the predominant devices utilized for the modulation process in ONNs [131, 29, 130]for SLMs, and [68] for metasurfaces. However, the cost implications of these technologies are considerable. SLMs, for instance, require an investment of several thousand dollars each, making them a costly option for widespread industrial application. Similarly, metasurfaces necessitate precision fabrication using Printed Circuit Board (PCB) technology, which further escalates costs. These financial constraints significantly impede the feasibility of ONNs for industrial production, as the high initial expenses pose a challenge to achieving cost-effective scalability and integration within existing systems.

To overcome these challenges, future research will need to focus on developing more robust and adaptable optical components that can be easily integrated into existing electronic systems. This includes research into new materials that can enhance the performance and durability of optical components.

1.3 Contribution of the Dissertation

In the contribution section of this dissertation, we outline the significant advancements and novel insights offered by our research within the broader context of the field. The study makes several critical contributions to the research field, which can be summarized as follows:

1.3.1 Beamforming in RFID systems

Although billions of battery-free backscatter devices (e.g., RFID tags) are intensively deployed nowadays, they are still unsatisfying in the two major performance limitations (i.e., short reading range and high miss reading rate) resulting from the current harvesting inefficiency. The classic beamforming technique is regarded as the most promising solution to address the issue. However, applying it to backscatter systems meets the deadlock start problem, i.e., without enough power, the backscatter cannot wake up to provide channel parameters; but, without channel parameters, the system cannot form beams to provide power. In this work, we propose a new paradigm called *transfer beamforming* (TBF), namely, the beamforming strategies can be transferred from reference tags with known positions to power up other unknown neighbor tags of interest. In short, transfer beamforming (is accomplished) via (launching) beamforming (to reference tags first) for (the purpose of) transfer. To do so, we adopt the semi-active tags as the reference tags, which can be powered up with a normal reader in a wide range. Then the beamforming is initiated and transferred to power up the low-sensitive but cost-effective passive tags surrounded by reference tags. A prototype evaluation of TBF with 8 transmitting antennas presents a 99.9% inventory coverage rate in a crowded warehouse with 2,160 RFID tags. Our comprehensive evaluation reveals that TBF can improve the power transmission by 6.9 dB and boost the inventory speed by 2× compared with state-of-art methods.

1.3.2 Optical Neural Network

Deep learning excels in advanced inference tasks using electronic neural networks (ENN), but faces energy consumption and limited computation speed challenges. To mitigate this, optical neural networks (ONNs) were developed, utilizing light for computations. However, their high manufacturing costs limited accessibility. In this work, we first introduce the binary optical neural network (BONN) a streamlined ONN

variant with binarized weights, which significantly reduces fabrication complexities and costs. Specifically, we address (i) the development of a binarization weight function aligned with backward-error propagation, and (ii) a simulation-based training for extra-large neural networks housing millions of neurons. We prototype six BONNs, each comprising four $0.8 \times 0.8mm^2$ layers with one million 800 nm diameter neurons. Costs are cut to 0.13 USD per layer, marking a substantial decrease of $769\times$ from previous ONNs. Experimental results reveal BONNs consume $2,405\times$ less power than leading ENNs while maintaining an average recognition accuracy of 74% across six datasets.

1.4 Organization of the Dissertation

The organization of this dissertation is structured to provide a coherent flow of ideas and a systematic exposition of the research. This thesis is organized into several chapters, each focused on a specific aspect of beamforming systems. Here is an overview of each chapter:

Chapter 1 gives a brief introduction to the beamforming system, including the advanced developments and sigmificant applications. Chapter 2 lays the foundational knowledge of beamforming, illustrating the principles of communication and beamforming technologies. Chapter 3 goes into optical neural networks, which is an important application of beamforming in optical computing, detailing the principles of diffraction and light propagation essential to these systems. Chapter 4 introduces an innovative approach to blind beamforming that utilizes the spatial relationships among RFID tags to enhance specificity and efficiency in target regions. Chapter 5 discusses a binary optical neural network that employs a differentiable binary method to drastically reduce production costs and shorten training time with a simplified light propagation calculation. The final chapter synthesizes the dissertation's findings, discussing the advantages and limitations of the proposed systems and suggesting future

research directions to extend and refine these technologies.

Chapter 2

Background of Beamforming system and Optical Neural Network

In this chapter, we will go into the working principles of the beamforming system and optical neural network to facilitate a deeper understanding of the background in subsequent sections.

2.1 Background of Beamforming system

Initially, we will explore the hardware structure of RFID systems, delving into their fundamental components and operational mechanisms. Following that, we will illustrate the working principles of beamforming, providing a detailed analysis of its theoretical foundation, practical implementations, and its role in enhancing the performance and efficiency of RFID systems.

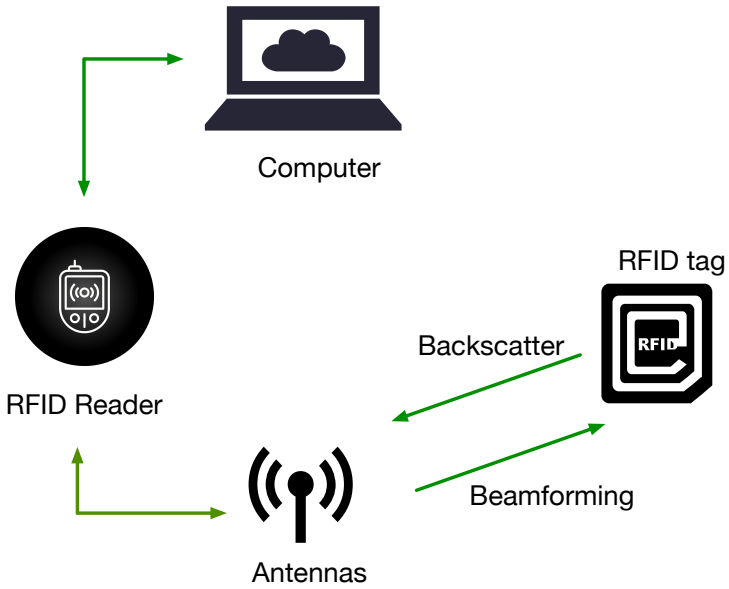


Figure 2.1: RFID system

2.1.1 Structure of RFID Systems

Beamforming technology, traditionally utilized in various communication systems, has been effectively adapted to operate within RFID communication systems, where it significantly enhances signal clarity and precision. This adaptation allows beamforming to focus radio frequency signals more precisely on specific RFID tags, thereby optimizing the system's performance by improving the accuracy and range of data transmission.

Fig. 2.1 is a typical RFID system, including a computer for data processing, an RFID reader, antennas, and RFID tags. Each part serves a specific function in the overall operation of the system.

RFID tags carry the information associated with the items to which they are attached. They function as the data-carrying component of the system. The tags microchip stores the information, such as serial numbers, price, manufacturing details, and other relevant data. When activated by a reader's signal, the tag responds by sending this data back to the reader. This process can be called "backscatter".

The antennas in an RFID system facilitate the transmission and reception of radio signals. They are connected to the RFID reader. They serve 2 main functions: 1. Emitting Radio Waves: Antennas send out radio waves that activate the tags, making them essential for starting the data transmission process. The beamforming technology will be utilized in this step to enhance the signal and improve the signal-to-noise ratio. 2. Receiving backscattered Signals from Tags: Once the tags backscatter their signal, the antennas capture these signals and convey them to the reader. The efficiency and range of data capture depend significantly on the antenna's design and placement relative to the tags.

The RFID reader is responsible for initiating communication with RFID tags. It sends out a radio frequency signal that powers passive tags and prompts all tags within range to transmit their stored data back to the reader. The reader then captures this data and relays it to the computer system for processing. The reader effectively serves as the intermediary between the physical items (tags) and the digital data system.

The computer system functions as the center for data aggregation, processing, and analysis. It collects the information received from the RFID reader and converts raw RFID data into usable information, categorizing and storing it appropriately.

In an RFID system, each component plays a critical role in ensuring the smooth and efficient tracking and management of items. RFID tags hold and transmit item-specific data; RFID readers and their antennas facilitate communication by emitting and receiving radio signals; and the computer system processes all incoming data to provide actionable insights. Together, these components form a powerful tool for inventory management, asset tracking, and data collection, driving operational efficiencies across numerous sectors.

2.1.2 Working Principle of Beamforming

Beamforming integrates antenna technology with digital signal processing to direct signal transmission or reception in specific directions. At the receiver, signals from multiple antenna elements are weighted and combined to construct the desired signal pattern. This process effectively forms a focused beam toward a specified direction. For effective beamforming, the use of multiple antenna systems is crucial. For instance, in Multiple Input Multiple Output (MIMO) systems, both multiple receiving and transmitting antennas are employed. This setup allows wireless signals from the transmitter to the receiver to follow multiple spatial streams across various paths, enhancing the signal-to-noise ratio significantly through sophisticated algorithms that process the signals from these multiple antennas. Figure 2.2 illustrates a simple MIMO structure with M transmitting antennas and N RFID tags. The transmission channel between the i^{th} transmitting antenna and the j^{th} receiving tag is denoted by $h_{i,j}$. It is assumed that all transmitting antennas are synchronized to the same baseband, thus sharing the same frequency but differing in phase and amplitude. The signal received by the j^{th} tag from multiple transmitting antennas is expressed as follows:

$$S_j(t) = \sum_{i=1}^{M} h_{i,j} a_i e^{J(2\pi f t + \phi_i)} = \sum_{i=1}^{M} a_{i,j} a_i e^{J(2\pi f t + \phi_i + \phi_{i,j})} \tag{2.1}$$

In this expression, J represents the imaginary unit, f denotes the frequency of the transmitted signal, a_i and ϕ_i are the amplitude and phase of the transmitting signal, respectively. The channel formula, $h_{i,j} = a_{i,j} e^{J\phi_{i,j}}$, depends on the path between the transmitted signal and the receiving tag. It incorporates both the amplitude $a_{i,j}$ and the phase shift $\phi_{i,j}$ associated with the i^{th} transmitting antenna and the j^{th} receiving tag.

The goal of beamforming is to create constructive interference towards the desired direction and destructive interference in others, as illustrated in Fig. 2.3. This results

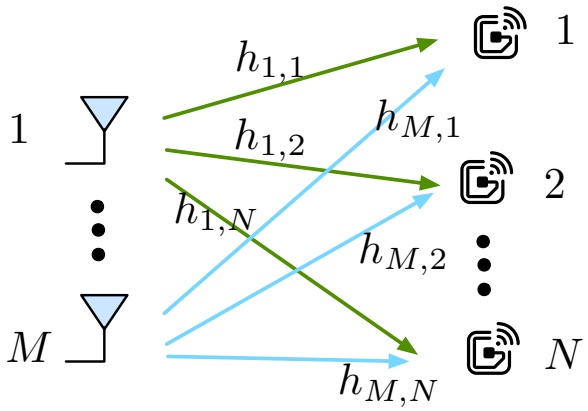


Figure 2.2: MIMO Structure

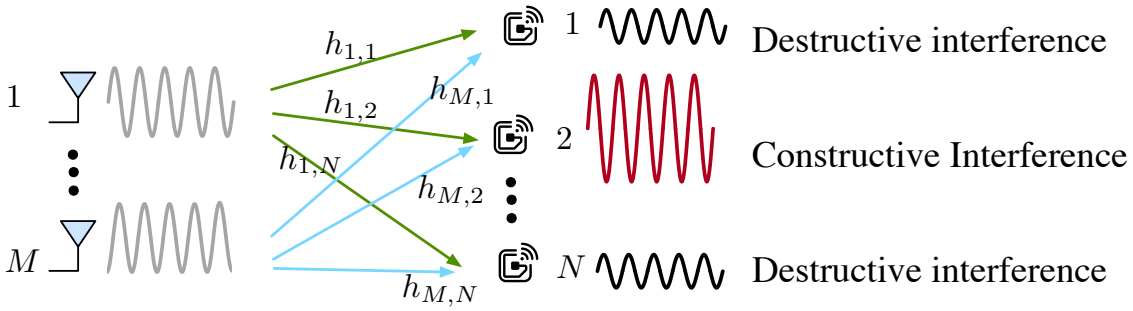


Figure 2.3: Beamforming for MIMO

in signal superposition specifically aimed at the desired target, which is tag 2. The signals at other tags are depressed. How is this achieved?

According to Euler's formula, Eqn 2.1 can be expanded as:

$$S_j(t) = \sum_{i=1}^{M} a_{i,j} a_i \cos(2\pi f t + \phi_i + \phi_{i,j}) + J \sum_{i=1}^{M} a_{i,j} a_i \sin(2\pi f t + \phi_i + \phi_{i,j}) \tag{2.2}$$

Let's assume that the direction to tag j is the desired direction. In this case, the amplitude of $S_j(t)$ should be maximized , meaning the expression $\sum_{i=1}^{M} a_{i,j} a_i \cos(2\pi f t + \phi_i + \phi_{i,j})$ should reach its maximum value. Assuming that the channel conditions remain relatively stable, the adjustment of the transmitting antennas' phase set $[\phi_1, \phi_2, \ldots, \phi_M]$ becomes the primary method for maximizing the amplitude of the re-

ceived signal. The objective of beamforming is to find a set of phases $[\phi_1, \phi_2, \ldots, \phi_M]$ that meet this requirement:

$$\forall i \in 1, 2, ..., M, \phi_i + \phi_{i,j} = n * 2\pi, n \in \mathbb{Z} \tag{2.3}$$

Then the amplitude of the received signal can be simplified as:

$$S_j(t) = \sum_{i=1}^{M} a_{i,j} a_i \cos(2\pi f t + \phi_i + \phi_{i,j}) = \sum_{i=1}^{M} a_{i,j} a_i \cos(2\pi f t) \tag{2.4}$$

The exploration of beamforming technology reveals its critical role in enhancing communication systems by directing signals towards specific targets while minimizing interference elsewhere. By adjusting the phases of signals from multiple antennas, beamforming ensures that signals are constructively combined at the desired location, thereby maximizing efficiency and signal clarity. This capability is invaluable in environments where precision and reliability are paramount, making beamforming a key player in advancing modern communication. Acquiring the channel information $\phi_{i,j}$ proves challenging due to the complexities and dynamic nature of the environments. This complexity makes it difficult to accurately determine the phases of the transmitted signals $[\phi_1, \phi_2, \ldots, \phi_M]$. The dynamic environmental factors continually alter the characteristics of the channel, requiring adaptive strategies to accurately estimate and compensate for these phase shifts in real-time applications.

2.2 Background of Optical Neural Network

Optical Neural Networks (ONNs) use the error backpropagation algorithm for beamforming because finding the optimal solution for thousands of passive transmission nodes is challenging with traditional algorithms. Besides beamforming, ONNs also rely on light diffraction to function. While the principles of beamforming were discussed in a previous chapter, this chapter will focus on the diffraction principle.

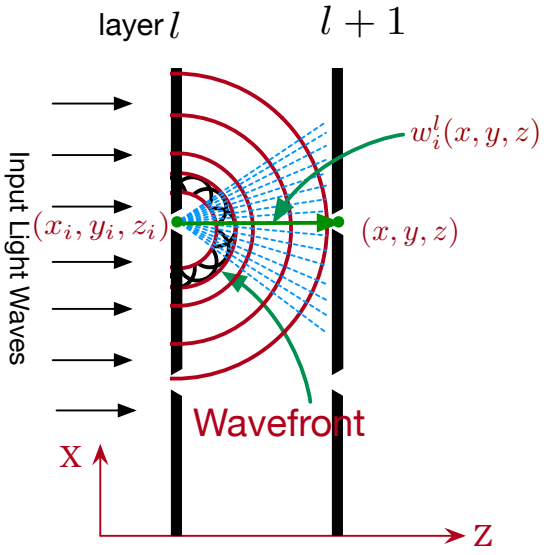


Figure 2.4: Diffraction Model

Diffraction is a core phenomenon of optics that allows light to bend around obstacles and spread as it passes through apertures. This background will delve into how ONNs harness diffraction, particularly through the Rayleigh-Sommerfeld diffraction equation, to perform complex computations that are inherently parallel and extremely fast. This discussion aims to elucidate the theoretical foundations of diffraction within ONNs, providing a clear view of their capabilities and the future possibilities they hold.

In optics, diffraction is critical in determining how light waves spread as they pass edges or through narrow apertures. The fundamental principle underlying diffraction is that every point on the wave surface could be regarded as the wave source of the emitted secondary wave, emitting spherical secondary wave respectively. The envelopment surface of these secondary waves would be the new wave surface. This is central to implementing neural network functions in ONNs. The diffraction forms the connection between two adjacent layers as shown in Fig. 2.4. When the input light waves encounter obstacles for example, at (x_i, y_i, z_i), then the light will begin to diffract. Each point on the wavefront will be a secondary source to produce subwaves,

the wavefront at any subsequent moment is the envelope of all these sub-waves. Then the signal is propagated from layer l to adjacent layer $l+1$ in this way. And the neurons between 2 adjacent layers are connected. We assume the position of neuron i on layer l is (x_i, y_i, z_i), the weight connected to neuron (x, y, z) on layer $l+1$ is defined as:

$$w_i^l(x, y, z) = \frac{z - z_i}{r^2}(\frac{1}{2\pi r} + \frac{1}{J\lambda})e^{\frac{2\pi r}{\lambda}} \tag{2.5}$$

where $r = \sqrt{(x_i - x)^2 + (y_i - y)^2 + (z_i - z)^2}$ represents the distance between the two neurons, $J = \sqrt{-1}$. The output of neuron (x_i, y_i, z_i) on layer l is then given by:

$$n_i^l(x_i, y_i, z_i) = w_i^l(x, y, z)t_i^l(x_i, y_i, z_i)\sum_{k=1}^{M} n_k^{l-1}(x_i, y_i, z_i) \tag{2.6}$$

where $\sum_{k=1}^{M} n_k^{l-1}(x_i, y_i, z_i)$ is the sum of the inputs to neuron (x_i, y_i, z_i) from the previous layer $l-1$, and $t_i^l(x_i, y_i, z_i)$ is the modulated transmission coefficient which is a complex number, which is defined as follows:

$$t_i^l(x_i, y_i, z_i) = a_i^l(x_i, y_i, z_i)e^{j\phi_i^l(x_i, y_i, z_i)} \tag{2.7}$$

If the modulation involves only amplitude, then $\phi_i^l(x_i, y_i, z_i)$ remains constant. Conversely, if the modulation is solely based on phase, then $a_i^l(x_i, y_i, z_i)$ should ideally remain constant, typically set to 1.

By carefully modulating the amplitude and phase of light, ONNs can mimic the dense network connections of their electronic counterparts, allowing for complex computations similar to those in fully connected layers. This approach not only leverages the high-speed and parallel processing advantages of optical systems but also aligns closely with established neural network methodologies, offering a promising pathway to enhance data processing capabilities in various technological applications.

After establishing the connection, we will utilize the error back-propagation algorithm to optimize the transmission coefficient $t_i^l(x_i, y_i, z_i)$ of each trainable node. The

weight $w_i^l(x, y, z)$ remains fixed once the node's position is determined. Following N modulation layers, the CCD captures the intensity of the resulting complex optical field, which is characterized as follows:

$$s_i^{N+1} = \left| \sum_{k=1}^{M} n_k^{N+1}(x_i, y_i, z_i) \right| \tag{2.8}$$

The primary objective of Optical Neural Networks (ONNs) is beamforming towards a target region. Given that the desired output is known, the error is defined as follows:

$$E = \frac{1}{K} \sum_{k=1}^{K} (s_k^{M+1} - \hat{s}_k^{M+1}) \tag{2.9}$$

where K represents the number of nodes of the output layer and $\hat{s}$ is the desired output. We can utilize the chain rule to calculate the gradient of the learnable transmission coefficients t. These parameters are then updated iteratively until the error E converges. This method ensures continuous optimization to achieve the most accurate model output.

Chapter 3

Transfer Beamforming via Spatial Relationships

3.1 Motivation

A long-standing vision for ultra-low-power ubiquitous sensor networks is for numerous tiny wireless sensors (e.g., smart dust [56]) to be embedded into every object. The emerging battery-free backscatter communication technology is regarded as the most promising solution to achieve this end. Unlike traditional active wireless networks, passive backscatter nodes harvest energy from the surrounding RF signals with μW-level power consumption, With such remarkable energy efficiency, various backscatters (e.g., RFID, LoRa backscatter [105], WiFi backscatter [59], etc) are proposed and being employed in diverse IoT applications.

A prominent backscatter system consists of an RF source, passive backscatter nodes, and active receivers. The RF source is the fundamental infrastructure that provides wireless charging service to nearby backscatter nodes. After being powered up, backscatter nodes transmit data to the receiver by using backscatter communication, that is, reflecting or not-reflecting the incoming RF signal that represents the bit one

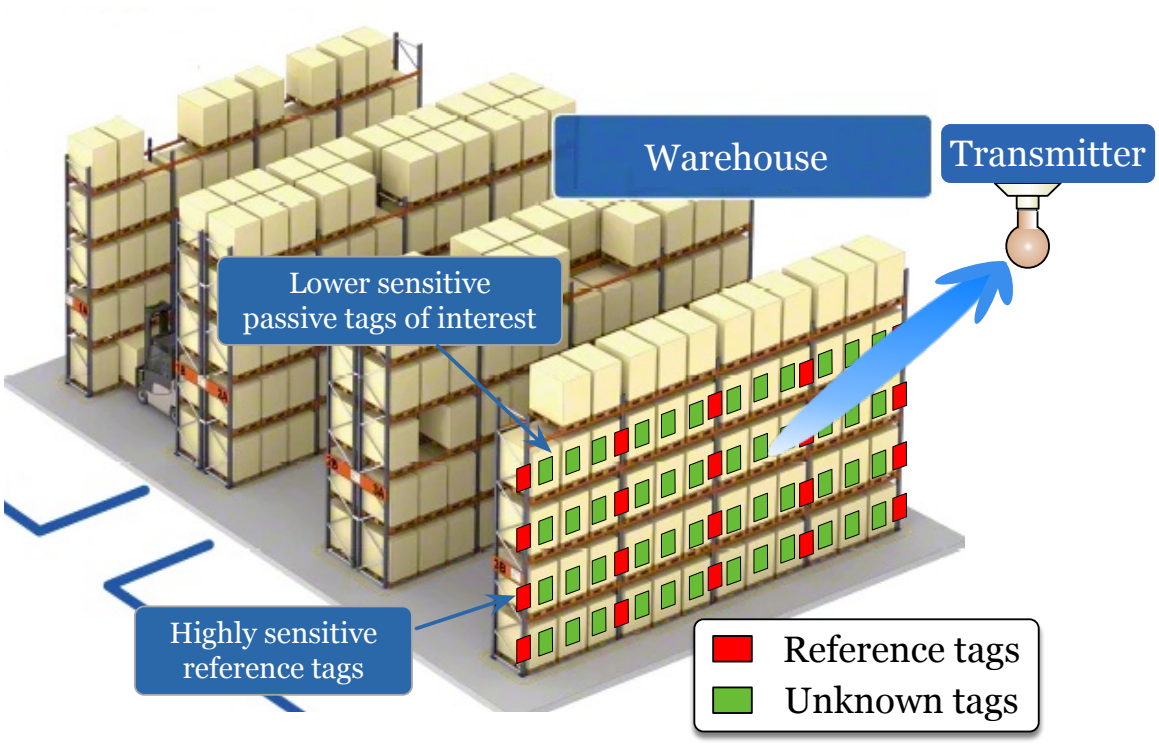


Figure 3.1: Applying beamforming systems in warehouse-like scenarios. Highly sensitive reference tags are attached on the shelves with known positions, while massive sensitive and cost-effective passive tags are placed on the shelves. Our goal is to choose appropriate beamforming strategies to power up all unknown tags with the help of reference tags.

or zero. As the pioneer of backscatter communication, UHF RFID is currently the most mature commercial off-the-shelf (COTS) backscatter system, in which the RF source and gateway are combined into a single device called a *reader* and backscatters are referred to *tags*. Nowadays, RFID systems have been widely adopted in various applications, ranging from retail management to asset tracking.

Nevertheless, current COTS RFIDs and the newly emerging backscatters still have two main limitations:

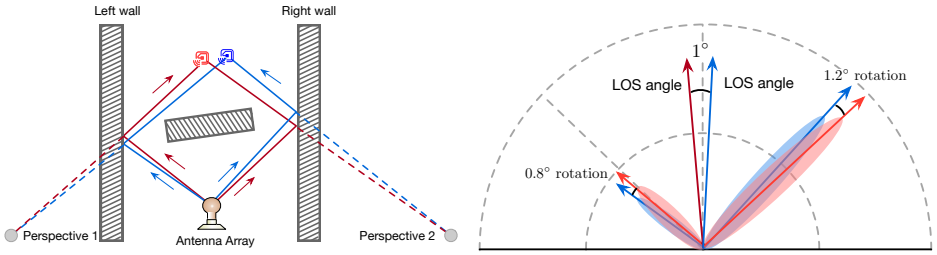
- **Limited range**. An RFID tag is successfully powered up at a distance of 5∼

15 m given a 1-watt output power at the RF source even if the backscattered signals can be detected hundreds of meters away. This bottleneck is caused by the harvesting inefficiency of miniature antennas equipped to a tag and the relatively higher threshold of minimum power to activate the IC (i.e.,-30 dBm [35]). As a result, RFID-tagged items can be detected only around the specific checkpoints in the vicinity of an RFID reader [51].

- **Miss readings**. Today's UHF RFID is unable to achieve the 99.9% accuracy needed by complex logistic networks where the propagations of RF signals are reflected unexpectedly. Consequently, about 10% areas called *blind spots* still exist, which cannot be covered with sufficient power to activate the tags due to the deconstructive superimposition of these coherent reflections [71, 22].

To address the above issues, the research community resorts to the classic beamforming technique. Beamforming utilizes an antenna array to concentrate the power on some specific regions through constructive interference while ensuring the total transmitting power in accordance with FCC regulations on the maximum power [27]. Notably, beamforming can well address the above limitations. First, it can raise the receiving power at tags, thereby extending the communication range and a single reader's coverage. Second, the RF source can deliver power from multiple directions concurrently, producing diverse interference of RF signals in the same region, which greatly reduces the presence probability of blind spots.

However, applying beamforming to a backscatter system faces a peculiar *deadlock start* that active radio systems never encounter. In an active radio system, the gateway (e.g., WiFi AP or base station) can request active nodes (e.g., mobile phone) to transmit short signals to acquire the channel state information (CSI). The CSI can help the gateway adopt the appropriate beamforming parameters. Unfortunately, this method fails in backscatter systems because of the deadlock problem: *Without enough power, the backscatter cannot wake up to provide CSI, but without CSI, the system cannot form beams to provide power.* The only approach is to search the



(a) Scenario (b) Beam profile

Figure 3.2: Intuition underlying transfer beamforming. (a) Antenna array uses two RF beams reflected off walls to power up two nearby tags respectively. (b) The beam profile used to power up the red tag can be transferred to power up the blue tag by rotating the RF beams by some degrees.

entire beamforming space exhaustively to wake up all backscatters [81, 82], resulting in a time-consuming cold start. To address this challenge, previous work either assumed to target a limited region (in-body [31, 38, 113]) or consume a wider band for random interference [74, 16].

In this work, we introduce the design and implementation of *transfer beamforming* (TBF), a general-purpose beamforming technology that aims to resolve the deadlock start in a backscatter system (e.g., RFID systems) by taking advantage of reference tags. We consider the general warehouse-like scenarios where RFID tagged items are highly clustered in a container (e.g., a shelf, a case, a pallet, etc), as shown in Fig. 3.1. Such locality and clustering features inspire us to pre-deploy some reference tags at known positions on the shelves. Suppose we have prior knowledge about how to form RF beams to light up these reference tags, we can then transfer the RF beams to the neighboring unknown tags of interest. The intuition behind transfer beamforming is that signals of nearby tags are received along closer paths when being reflected off each surface. This phenomenon has been already observed and widely used for RFID localization [117, 84]. Unlike previous work, we leverage this phenomenon to develop the beamforming strategy.

To illustrate our approach, Fig. 3.2(a) shows a toy example with two tags, where the two tags (red and blue tags) are separated by 10 cm. The antenna array can still power up the two tags by using the reflections off walls even if the signals in the line of sight (LoS) of the two tags are blocked. To this end, the antenna array must steer two RF beams toward two directions for each tag as shown in Fig. 3.2(b). This figure shows the *beam profile*, which describes how much RF power is allocated to a direction. Notably, the red and blue profiles are similar. If the beam profile to power up the red tag is known, then we can transfer it to power up the blue tag by rotating the two red beams by about 1° (i.e., the difference of two LoS angles) counterclockwise. Translating these intuitive opportunities into concrete gains entails challenges.

• First, how can we acquire the optimal beamforming strategies for reference tags? To do so, we adopt semi-active RFID tags as references. These battery-powered and long-life (e.g., 10 years) tags are equipped with highly sensitive frontends, and thus can be woken up in a wide-range warehouse without using beamforming. With the traditional feedback-based beamforming (FEB) approach, we can estimate the beamforming strategies (i.e., beamformer configurations) which can light up the positions of reference tags.

• Second, how can we transfer a known beam profile to light up a neighboring position? Given a beam profile that can light up a position P_1, we should rotate the RF beams to light up an unknown neighbor position P_x. First, we must compute the orientation of P_x relative to the P_1. Then, we compute the angle of rotation based on the directions of two positions relative to the antenna array. Finally, we rotate RF beams in the beam profile by the same degrees and in clockwise or counterclockwise directions to produce the transferred beam profile.

• Third, how can we develop the beamforming strategy for tags of interest at unknown positions? We visit each possible grid surrounded by reference tags. In each grid, we find the K-nearest reference tags, transfer their beam profiles and fuse them into a

single one, which is further reversed to a beamforming strategy by using a trained deep neural network. Eventually, the array takes this strategy to light up this unknown position and identify the tags, if present. To speed up the searching, we also adopt pruning optimization to balance the trade-off between the optimal strategy and the performance.

Contributions. To the best of our knowledge, this work is the first to propose transfer beamforming for backscatter systems. A prototype evaluation on an eight-antenna reader shows TBF can achieve a 99.9% coverage rate in a crowded warehouse with 2,160 commercial RFID tags. Compared with state-of-the-art works, TBF can boost the RF power transmission by 6.9 dB and the inventory speed by 2×.

3.2 Background

In this section, we review the background of our beamforming system and then formalize the beamforming problem.

3.2.1 Beamforming System

In this section, we follow [28] and provide background on the beamforming system in the full-duplex setting. Beamforming is a technique that can concentrate a wireless signal toward a specific direction by using an antenna array, rather than spread the signal in all directions as it normally would. Fig. 3.3 shows an N-element full-duplex beamforming system, where each antenna is shared by a pair of Tx and Rx elements via a circulator. Each Tx element consists of a power amplifier (PA) and a phase shifter (PS). Symmetrically, each Rx element consists of a low-noise amplifier (LNA) and a PS. The PS can shift the phase of the signal with a continuous value within $0 \sim 360°$. The copies of the baseband signal are propagated into the air via their antennas. These in-air RF signals operate at the same frequency and carry the

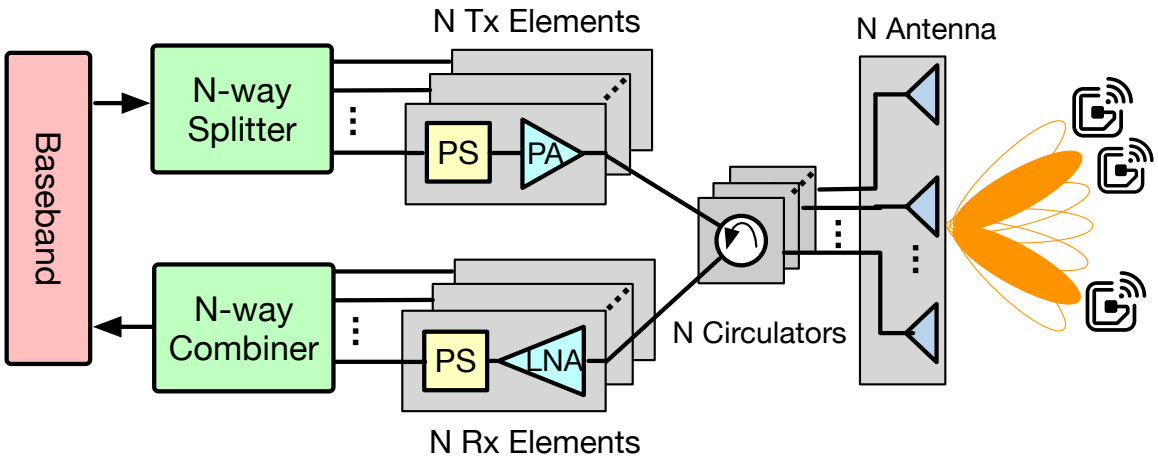


Figure 3.3: An N-element beamforming system.

same data, but differ in phase values. As a result, these RF signals will produce constructive interference in some areas and destructive interference in other areas. The constructive (or destructive) interference leads to stronger (or weaker) signals received in those areas.

Thus, the core of beamforming is the selection of a group of appropriate phase values for the beamformers called *beamforming configuration* or *beamforming strategy* to concentrate the RF signals at a desired angle or region (due to reflections) where the receiving devices are located. Suppose our RF source is equipped with an M-antenna array. The question then is how many beamforming strategies can we develop? In theory, the beamforming system can generate any continuous phase shift. Actually, it is still limited to the number of bits representing a phase. Let L_{PS} denotes the number of phase shifts that a PS can generate. Then, the total number of beamforming strategies is up to $\mathcal{O}(M^{L_{\text{PS}}})$. When $M = 8$ and $L_{\text{PS}} = 16$ are used, an astonishing $8^{16} = 2.8 \times 10^{14}$ configurations are produced.

3.2.2 Problem Formularization

Let S denote the RF signal transmitted from a transmitting antenna and received at a tag. Then, it can be given as follows:

$$S(t) = h \cdot e^{\mathbf{J}(2\pi ft+\phi)} = \tilde{a}e^{\mathbf{J}(2\pi ft+\phi+\tilde{\phi})} \tag{3.1}$$

where $\mathbf{J}$ denotes the complex number, $h = \tilde{a}e^{\mathbf{J}\tilde{\phi}}$ is the channel parameter that is highly dependent on the tag's position. $\tilde{a}$ and $\tilde{\phi}$ are the amplitude attenuation and phase rotation over the propagation path, respectively. We normalize the amplitude of the transmitted RF signal to 1 for simplicity because a beamforming system usually fixes the output power at each transmitting antenna. ϕ is the phase shift of the output signal. With regard to M transmitting antennas, the RF signal received by the tag is given by

$$\begin{aligned} S(t) &= \sum_{i=1}^{M} S_i(t) = \sum_{i=1}^{M} \tilde{a}_i e^{\mathbf{J}(2\pi ft+\phi_i+\tilde{\phi}_i)} \\ &= \sum_{i=1}^{M} \tilde{a}_i \cos(2\pi ft + \phi_i + \tilde{\phi}_i) + \sum_{i=1}^{M} \mathbf{J}\tilde{a}_i \sin(2\pi ft + \phi_i + \tilde{\phi}_i) \end{aligned} \tag{3.2}$$

Notably, the constructive interference can be achieved at the tag only when the M sinusoidal waves align with each other. Formally, *we say M RF signals are aligned with each other when the following condition is fulfilled:*

$$(\tilde{\phi}_1 + \phi_1) \bmod 2\pi = (\tilde{\phi}_2 + \phi_2) \bmod 2\pi = \cdots = (\tilde{\phi}_M + \phi_M) \bmod 2\pi \tag{3.3}$$

Let $\tilde{\mathbf{\Phi}}$ and $\mathbf{\Phi}$ represent the phase shifts caused by the in-air propagations and the phase shifts configured at the M transmitting antennas. Namely,

$$\tilde{\mathbf{\Phi}} = [\tilde{\phi}_1, \tilde{\phi}_2, \cdots, \tilde{\phi}_M] \text{ and } \mathbf{\Phi} = [\phi_1, \phi_2, \cdots, \phi_M] \tag{3.4}$$

$\tilde{\mathbf{\Phi}}$ and $\mathbf{\Phi}$ are the channel parameter and the beamforming strategy, respectively. Then, the goal of the beamforming technique can be formalized as follows:

Problem 1 (Beamforming)**.** *Given an unknown beamforming parameter $\tilde{\mathbf{\Phi}}$, we are looking for the strategy $\mathbf{\Phi}$ to make the RF signals received at the tag align with each other, namely, $(\tilde{\mathbf{\Phi}}+\mathbf{\Phi}) \bmod 2\pi \approx c\mathbf{I}$ where c is some constant and $\mathbf{I}$ is the unit vector.*

Unlike this general beamforming problem, we consider a relaxed problem, that is, how can we transfer the beamforming strategy from one position to another position? We use $\mathbf{\Phi}(P)$ to denote the beamforming strategy to light up position P (i.e., a tag can be powered up if it presents at P). Similarly, our problem is formalized below:

Problem 2 (Transfer Beamforming)**.** *Given K known beamforming strategies, denoted by $\{\mathbf{\Phi}(P_1), \ldots, \mathbf{\Phi}(P_K)\}$, which can successfully light up the positions of $\{P_1, \ldots, P_K\}$, how can we find a beamforming strategy $\mathbf{\Phi}(\mathbf{P_x})$ to light up an unknown position P_x?*

Notice that we never make any assumption on the propagation model. The LoS propagation might be present or absent. The RF beams might be reflected to light up the destination.

3.3 Overview

Transfer beamforming orients to backscatter systems (e.g., RFID systems). As a running example, we mainly present the system in the context of a warehouse-like scenario. The transfer beamforming technique can be applied to other similar backscatter communication systems.

3.3.1 Scope

This work aims to develop a practical wireless beamforming system, with the goal of quickly concentrating energy on a large number of passive tags at unknown positions. Traditional beamforming systems can focus energy toward a specific direction. However, it cannot deal with occlusion and complex scenarios (such as a warehouse, library, supermarket, etc) when tags are blocked from the antenna array in the line of sight (LoS). Thus, modeling the beamforming via geometry is impossible even if the location of a tag is known. RF signals are reflected off surrounding objects like ceil-

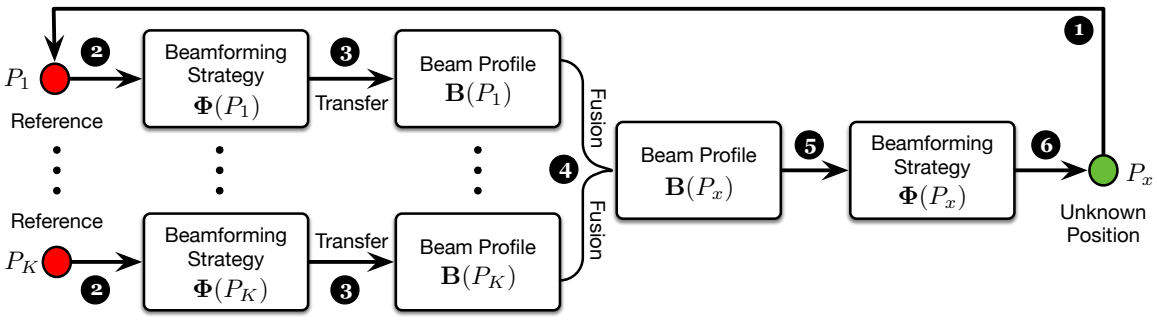


Figure 3.4: Workflow for transfer beamforming

ings and shelves in warehouse-like scenarios. In contrast, we take advantage of these reflections to identify the tags that cannot be powered up via the LoS RF beam. The locations of reference tags remain unchanged and known to the antenna array in advance. The unknown tags of interest inside the containers are surrounded by reference tags. These tags might be moved in or out when the related items are transferred. In Fig. 3.1, the reference tags are fixed on the shelf with a regular pattern, whereas the unknown tags are scattered on the shelves at random. Finally, all tags transmit their signals in a framed ALOHA protocol, so we do not consider the collided signals.

3.3.2 In a Nutshell

The intuition underlying the transfer beamforming is that nearby tags experience a similar multipath environment (e.g., reflectors in the environment). This phenomenon is not new and it has been used for RFID localization [117], which claims that two tags should be located closely if their signals are received from similar directions. Unlike previous work, we take the backward inference, that is, nearby tags can be powered up using a similar beam profile. In short, *given prior knowledge about beamforming strategy for a tag, we can infer a strategy to power up its nearby tags.* In other words, we can transfer the beamforming strategy from one tag to other nearby tags.

The high-level workflow is shown in Fig. 3.4. We go through the following steps for the transfer beamforming:

- First, when visiting an unknown position P_x surrounded by reference tags, we select K nearest neighbor reference tags, which locate at positions of $\{P_1, \ldots, P_K\}$ respectively.
- Second, we use the traditional FBB to sort out the optimal beamforming strategies for these K reference tags (i.e., $\{\mathbf{\Phi}(P_1), \ldots, \mathbf{\Phi}(P_K)\}$), as described in §3.4.
- Third, the corresponding K beam profiles (denoted by $\{\mathbf{B}(P_1), \ldots, \mathbf{B}(P_K)\}$) are generated from the above K strategies. They are rotated and merged into a single beam profile $\mathbf{B}(P_x)$, which lights up position P_k. The technique's details are described in §3.5.
- Finally, we reversely generate the beamforming strategy $\mathbf{\Phi}(P_x)$ from $\mathbf{B}(P_x)$ using the technique described in §3.6 and apply it to the antenna array to light up position P_x. If an unknown tag presents there, it is identified.

The above steps are repeated until all shelves or containers are visited. To speed up this process, we propose the pruning optimization approach(§3.6.2) to improve the beamforming efficiency. Notably, we can benefit from reference tags in two factors. First, we visit the space surrounded by reference tags instead of the entire warehouse because of the clustering nature of the warehouse-like scenario. Second, the beamforming strategies are inferred from the channels of reference tags, which can reflect the environmental dynamics in a timely manner.

3.4 Form Beams From Reference Tags

In this section, we introduce how the beamforming strategies for pre-deployed reference tags can be acquired.

3.4.1 Semi-active Tags as Reference

We adopt the semi-active tags as reference tags. Semi-active tags are based on the same principle as passive tags but include the battery, which helps increase many capabilities, e.g., communication range, activation sensitivity, etc. Semi-active tags also follow the Gen 2 protocol as passive tags and are interoperable with current passive Gen 2 readers. Semi-active tags remain dormant to reduce energy consumption until a reader sends a wake-up signal. Such an energy-saving mechanism enables them to work for several years, even up to 10 years [35]. However, semi-active tags cannot replace passive tags for large-scale deployment because their prices are higher than passive tags.

We prefer semi-active tags as references because they provide longer communication ranges and high sensitivity in detecting the reader signal. Usually, a passive tag can be powered up by receiving the power of -10 dBm or above (i.e., the sensitivity) but the sensitivity is lowered to -40 dBm in semi-active tags due to the battery-assisted RF circuity. Thus, we can easily wake up all semi-tags in a large warehouse with a single omnidirectional antenna.

3.4.2 Acquiring Strategy for a Reference Tag

We use another single-antenna reader (e.g., Impinj R420) to wake up all reference tags because of the higher sensitivity and to dispense with beamforming. After waking up, the reference tags respond with their electronic product codes (`EPC`s). During this process, we use the antenna array to sniff the backscatter signals from reference tags. Suppose a reference tag at position P transmits its signal. Let $\tilde{\phi}_{P \to A_i}$ denote the phase of the signal received at the antenna A_i of the array. $\tilde{\phi}_{P \to A_1}$ is the phase shift caused by the in-air propagation from $P_1 \to A_1$. We can estimate M phase values as follows:

$$\{\tilde{\phi}_{P \to A_1}, \tilde{\phi}_{P \to A_2}, \ldots, \tilde{\phi}_{P \to A_M}\} \tag{3.5}$$

Now, if constructive interference via beamforming is desired to appear at position P using the antenna array, what beamforming strategy should be adopted? Before answering this question, we first introduce the channel reciprocity:

Theorem 1 (Channel Reciprocity). *When an RF signal is transmitted from position $P_1 \rightarrow P_2$, the phase is shifted by ϕ caused by the in-air propagation. Then, the phase is also shifted by ϕ when it is transmitted from $P_2 \rightarrow P_1$.*

The channel reciprocity holds true regardless of whether NLOS propagation presents. Thus, when an RF signal is transmitted from A_i to position P, the phase is also shifted by $\tilde{\phi}_{P\rightarrow A_i}$ in the air because $\tilde{\phi}_{P\rightarrow A_i} = \tilde{\phi}_{A_i\rightarrow P}$. Thus, the optimal beamforming strategy is given as follows:

$$\mathbf{\Phi}(\mathbf{P}) = \{2\pi - \tilde{\phi}_{P\rightarrow A_1}, 2\pi - \tilde{\phi}_{P\rightarrow A_2}, \ldots, 2\pi - \tilde{\phi}_{P\rightarrow A_M}\} \tag{3.6}$$

where the optional phase shift introduced at the i^{th} beamformer is the negative of the received phase, i.e., $\phi_i = -\tilde{\phi}_{P\rightarrow A_i}$. A physical phase shifter does not support the negative offset so 2π is added for wrapping. As mentioned earlier, constructive interference is achieved only when the M RF signals are aligned with each other at the destination. Now, the initial phase of the RF signal transmitted from $A_i \rightarrow P$ is $\phi_i = 2\pi - \tilde{\phi}_{P\rightarrow A_i}$; and the in-air phase shift is $\tilde{\phi}_i = \tilde{\phi}_{A_i\rightarrow P} = \tilde{\phi}_{P\rightarrow A_i}$. Consequently, the final phase of the RF signal from $A_i \rightarrow P$ is given by

$$\phi_i + \tilde{\phi}_i = 2\pi - \tilde{\phi}_{P\rightarrow A_1} + \tilde{\phi}_{P\rightarrow A_i} = 2\pi \tag{3.7}$$

The above equation can be extended to other antennas. Thus, we have the following equations:

$$(\tilde{\phi}_1 + \tilde{\phi}_1) = (\tilde{\phi}_2 + \tilde{\phi}_2) = \cdots = (\tilde{\phi}_M + \tilde{\phi}_M) = 2\pi \tag{3.8}$$

The condition of Eq. 3.3 is fulfilled. Thus, $\mathbf{\Phi}(P)$ (Eq. 3.6) is an optimal beamforming strategy to achieve constructive interference at position P.

The above actually introduces the traditional beamforming solution, which has been widely adopted in an active wireless system. The device of interest (i.e., mobile phone)

is requested to broadcast a beacon signal first and then the gateway can correctly steer the RF beams toward the device based on the estimated phase. We call this process *feedback based beamforming* (FBB). In this work, we use FBB for two purposes: First, FBB is used to let us know the optimal beamforming strategy to light up the positions where the reference tags locate. Second, FBB can help us find out the ground truth of strategy for unknown tags. Specifically, we use a commercial reader to power up an unknown tag at close range. The antenna array then listens to its signal and estimates the optimal beamforming strategy. However, FBB cannot resolve our problem unless all passive tags can be powered up somehow. Unfortunately, this situation will fall into the deadlock trap. In the following, we will use FBB to find out the ground truth to validate if the transferred beamforming strategy is correct.

3.4.3 Discussion

Acquiring the beamforming strategies for reference tags is greatly crucial. We do not need to power up them by beamforming. Instead, they provide us with an important clue about how to light up the positions that reference tags locate. Our next task will be to transfer the RF beams from these positions to potential locations that tags of interest might locate. The reference tags also help us tackle the environmental dynamics. When the environment is changed, such as when shelves are moved, new obstacles appear, or old ones disappear, we can update the beamforming strategies in a timely manner by acquiring the signals from reference tags regularly.

3.5 Transforming Beams

In this section, we address the issue of how to transform a beam profile acquired from a reference tag to light up an unknown position.

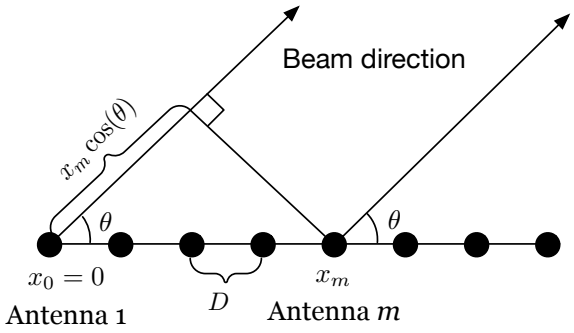


Figure 3.5: Beamforming with a standard uniform linear array. When the array steers its beam toward direction θ, the transmitted signal at the m^{th} antenna is projected to that direction with phase compensation.

3.5.1 Beam Profile from a Strategy

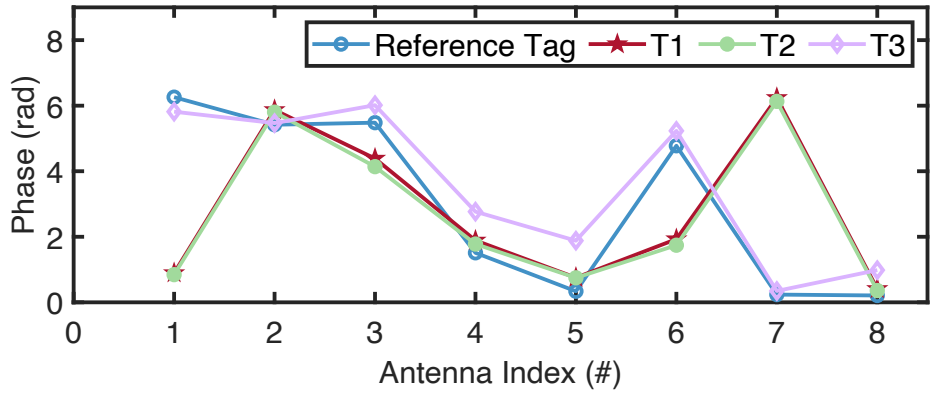
A beam profile is an immediate result of a beamforming strategy. After a strategy $\mathbf{\Phi} = \{\phi_1, \ldots, \phi_M\}$ is applied to the beamformers, the array will steer one or multiple RF beams toward some directions. We use the beam profile to quantify exactly how much power is concentrated in an arbitrary direction. We establish the standard mathematical formulation for a uniform linear array. Let S_m be the RF signal transmitted from the m^{th} antenna, $m = 1, \ldots, M$. $w_m(\theta)$ is the complex weight assigned to S_m if the RF beam is steered to the direction $\theta \in [0°, 180°]$. λ is the wavelength. D is the distance between two adjacent antennas where $D \leq \lambda/2$. The $x_m = (m-1)D$ is the position of the m^{th} antenna. The transmitting power concentrated toward the θ direction is computed as:

$$B(\theta) = |\sum_{m=1}^{M} w_m(\theta) \cdot S_m|^2 \tag{3.9}$$

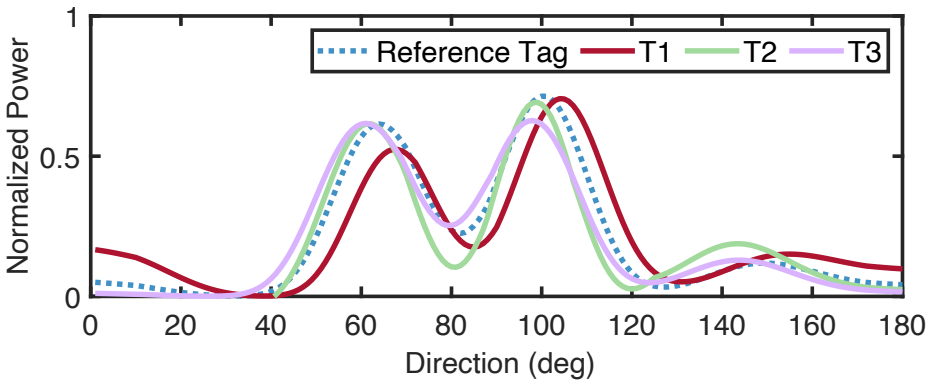
where

$$\begin{cases} w_m(\theta) = e^{-\mathbf{J}(2\pi x_m \cos(\theta)/\lambda)} \\ S_m = e^{\mathbf{J}(2\pi f t + \phi_m)} \end{cases} \tag{3.10}$$

where $\phi_m \in \mathbf{\Phi}$. At a high level, the $w_m(\theta)$ is a reverse phase compensation regarding the direction θ, as shown in Fig. 3.5. If $w_m(\theta)$ is aligned with the real direction,



(a) Beamforming Strategy



(b) Beam Profile

Figure 3.6: Illustration of the beam profile. (a) shows the optimal beamforming strategies to successfully power up a reference tag and its three neighbor tags; (b) shows the beam profiles generated from the strategies.

then the superimposition reaches the maximum. The above equation projects the transmitted signals to direction θ. By steering its beam across 180°, we can acquire the beam profile as follows:

$$\mathbf{B} = f(\mathbf{\Phi}) = \{B(0°), B(1°), \ldots, B(180°)\} \tag{3.11}$$

where $f(\cdot)$ represents the function that translates the strategy to a beam profile by using Eq. 3.9. Reversely, we can also translate a given beam profile to a beamforming strategy, which is represented by $\mathbf{\Phi} = f^{-1}(\mathbf{B})$.

To visually understand the beam profile, we use the FBB to acquire the optimal

beamforming strategies for one reference tag and four passive tags (i.e., Impinj H47). The three passive tags are 15 cm away from the reference tag. We show their optimal strategies in Fig. 3.6(a) for comparison. Visually, the four strategies take the same common configurations at some antennas(e.g., at A_2 and A_8). However, due to the periodicity of the phase value, the whole curves are not similar in appearance and fluctuate too much. This characteristic might cause the resultant direction of the RF beam to differ considerably even if the configuration at a beamformer is as small as 0.1°. Thus, we do not transfer beamforming based on the strategy directly.

Then, we generate their beam profiles based on the corresponding strategies by using Eq. 3.9. The results are shown in Fig. 3.6(b). The figure shows the power transmitted by the antenna array as a function of the direction. Each profile has two peaks, corresponding to the two major RF beams toward 60° and 100°, respectively. The figure successfully captures the fact that the beam profiles of the three neighbor tags are considerably similar to that of the reference tag. This example demonstrates two observations: First, the strategies might be different but their corresponding beam profiles are similar. Second, if two tags are close to each other in space, the RF beams in their profiles should closely match. This condition prompts us to consider that the transfer should be taken based on the beam profile instead of the strategy.

A beam profile indicates that the tag can be powered up only when the antenna array concentrates the power based on the profile. The resultant RF signal might be reflected off objects and finally arrive at the tag, as shown in Fig. 3.2(a). A beam profile itself cannot reveal the direct information about the reflections but implies it indirectly.

3.5.2 Transfer from a Single Profile

Imagine that the antenna array is currently steering its RF beams toward a tag at position P_1. Now, the tag moves to another nearby position P_2. The question here is

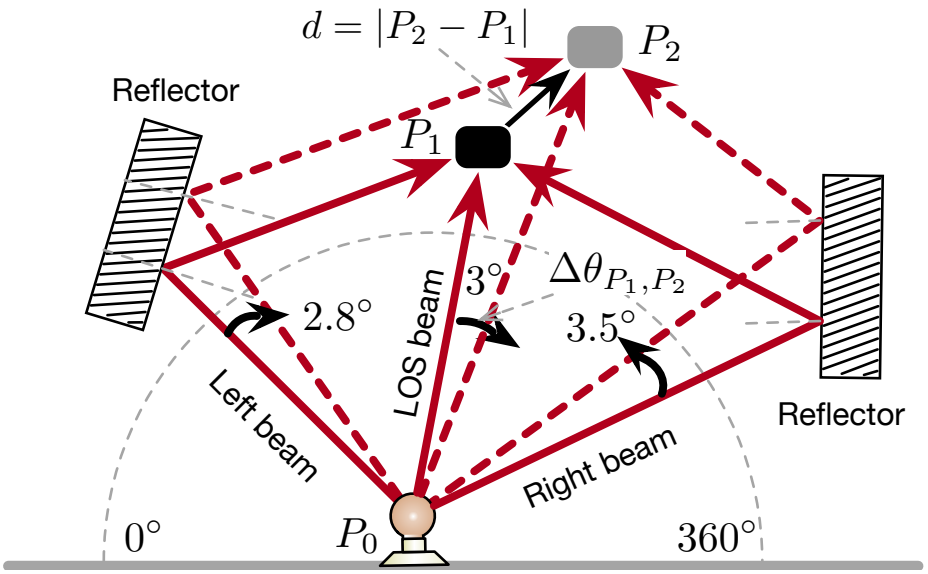


Figure 3.7: Illustration of transferring RF beams. All RF beams currently concentrate on position P_1. We can rotate the LoS beam 3° clockwise, the left beam 2.8° clockwise, and the right beam 3.5° counterclockwise to make the RF beams concentrate on position P_2.

how we could re-concentrate the RF beams on the tag? Fig. 3.7 shows this problem. When the tag is at P_1, three RF beams are focused on it, namely, LoS beam, left beam, and right beam. The LoS beam points to the direction of the LoS path. The angle of the left beam (or the right beam) is less (or greater) than the LoS angle. In the figure, the LoS beam must be rotated clockwise by 3° exactly, where P_0 is the center location of the array. Both the left and the right beams also need to rotate with similar degrees (e.g., 2.8° and 3.5°, respectively). However, the rotation directions for the two beams are completely different. The left beam is rotated clockwise, whereas the right beam is rotated counterclockwise due to the approximate mirror model. The direction of rotation depends on the orientation of P_2 relative to P_1. We have four types of orientations denoted by $O_{P_1 \to P_2}$, namely, P_2 is at the top left (↖), top right

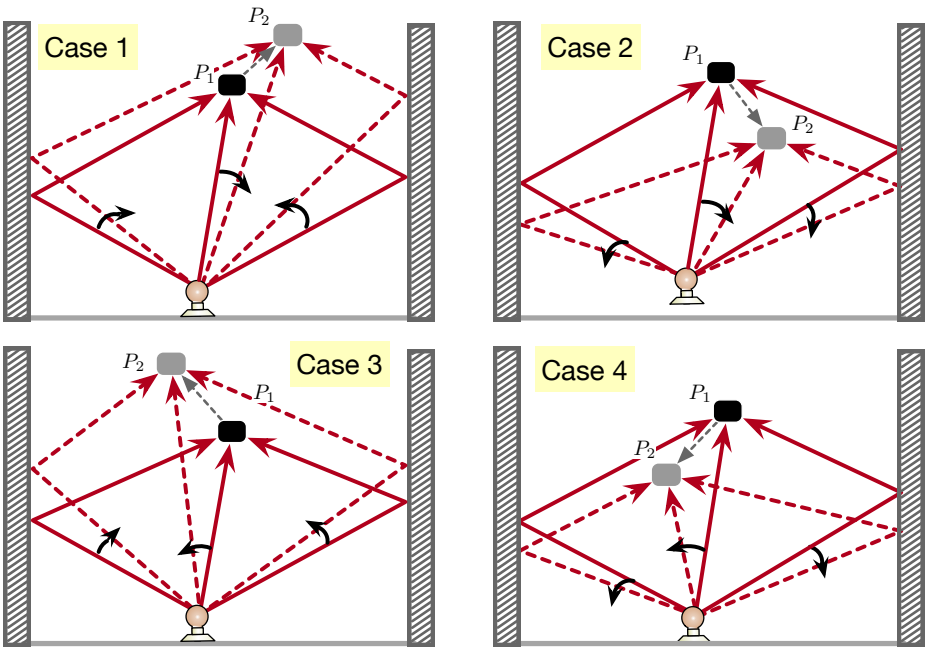


Figure 3.8: Four cases for transferring beams. The four cases consider position P_2 relative to position P_1. Case 1, 2, 3, and 4 show the P_2 is at the top-right, bottom-right, top-left, and bottom-left regions relative to P_1 respectively.

(↗), bottom left (↙) and bottom right (↘) of P_1.

$$O_{P_1 \to P_2} = \begin{cases} \text{Top Right}(\nearrow) & \text{if } |\angle(P_2 - P_1)| \in [0°, 90°) \\ \text{Top Left}(\nwarrow) & \text{if } |\angle(P_2 - P_1)| \in [90°, 180°) \\ \text{Bottom Left}(\swarrow) & \text{if } |\angle(P_2 - P_1)| \in [180°, 270°) \\ \text{Bottom Right}(\searrow) & \text{if } |\angle(P_2 - P_1)| \in [270°, 360°) \end{cases} \tag{3.12}$$

We show the geometric models for the four cases in Fig. 3.8 to provide a better understanding of the direction of the ratio. For example, in case 2, P_2 is at the bottom right of P_1. The LoS beam and the right beam are rotated clockwise but the left beam is rotated counterclockwise. We summarize the rotating directions of the four cases in Table 3.1.

The above model holds true regardless of the reflectors' orientations, material sizes, or positions because they exert similar impacts on P_1 and P_2 as long as they are spatially close. The reflections from the objects behind tags are not considered here

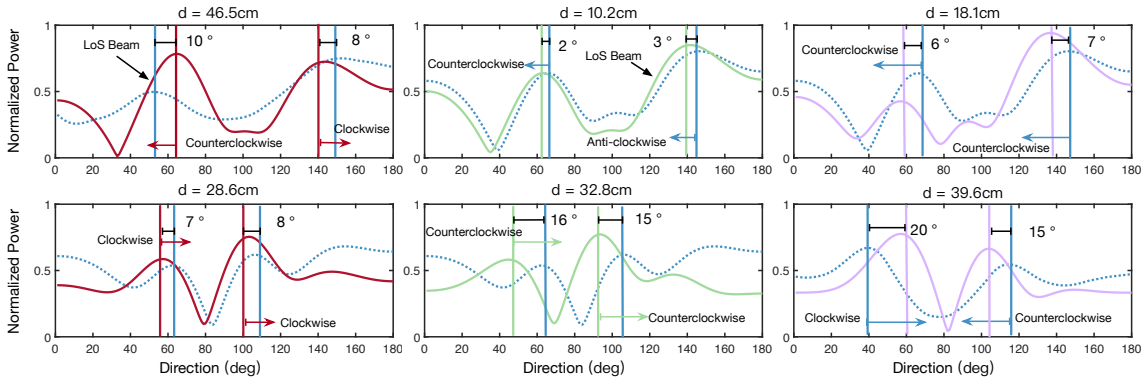


Figure 3.9: Beamforming transferability. We acquire the beam profiles from six pairs of nearby tags with different distances (d). Each figure compares the beam profiles for a pair of tags. They are highly similar and can be transferred to each other by shifting RF beams with a similar degree. However, the shifting direction might be different.

because they need longer propagations and become considerably weak. This situation also holds true even when the LoS path is blocked (i.e., the LoS beam does not exist) or there are additional obstacles in the NLOS path between the reader and tags. The beam to reference tags naturally contains information on reflections and obstacles in the environment. The paths with strong reflection are reused for the new beam and the paths with strong attenuation (i.e., obstacles) are removed. By strategically

Table 3.1: The rotating direction of RF beams. Three types of beams are rotated clockwise (i.e., shifting to right ($\rightarrow$) in the profile) or counterclockwise (i.e., shifting to left ($\leftarrow$) in the profile) based on the orientation of P_2 relative to P_1.

$O_{P_1 \rightarrow P_2}$	$\nearrow$	$\searrow$	$\nwarrow$	$\swarrow$
LOS Beam	$\rightarrow$	$\rightarrow$	$\leftarrow$	$\leftarrow$
Left Beam	$\rightarrow$	$\leftarrow$	$\rightarrow$	$\leftarrow$
Right Beam	$\leftarrow$	$\rightarrow$	$\leftarrow$	$\rightarrow$

placing reference tags that are aware of their positions, our system can infer the beamforming strategies that take into account not only the distance and angle but also the reflections and obstacles.

To verify the proposed transfer models, we choose six pairs of tags with different distances to acquire their optimal beamforming strategies (using FBB) and the corresponding beam profiles. The profiles are shown in Fig. 3.9. First, each pair has a similar beam profile as mentioned earlier. We also find that the similarity of the two profiles depends on two factors: the distance (i.e., d) and the direction difference (i.e., $\Delta\theta_{P_1 \to P_2}$), which are computed as follows:

$$d_{P_1 \to P_2} = |P_2 - P_1| \text{ and } \Delta\theta_{P_1 \to P_2} = |\theta_2 - \theta_1| \tag{3.13}$$

where $\theta_2 = \angle(P_2 - P_0)$ and $\theta_1 = \angle(P_1 - P_0)$ are the directions of the P_2 and P_1 relative to the array, respectively (Fig. 3.7). They are also called LoS angles. Smaller distance and direction difference will greatly improve the similarity. Second, RF beams in each profile might be shifted (i.e., rotated) to diverse directions. These experiments fully confirm the transferability of beamforming via the proposed transfer model.

Algorithm. Given a known beam profile $\mathbf{B}(P_1)$, which can successfully light up position P_1, we aim to synthesize the profile $\mathbf{B}(P_2)$ to light up the adjacent position P_2. We first compute the directions of P_1 and P_2 (i.e., θ_1 and θ_2) and their direction difference (i.e., $\Delta\theta_{P_1 \to P_2}$). Thus, the profile should be translated with $\Delta\theta_{P_2 \to P_1}$ degrees, i.e., the LoS beam is re-aimed from P_1 to P_2. Then, we compute the relative orientation $O_{P_1 \to P_2}$ by using Eq. 3.12. Next, the original profile $\mathbf{B}(P_1)$ is divided into several beam fragments, each of which contains a single RF beam (i.e., a peak). Visiting each RF beam, it can be categorized as a LoS beam, a left beam, or a right beam based on the orientation $O_{P_1 \to P_2}$. Next, each corresponding beam fragment is translated by $\Delta\theta_{P_2 \to P_1}$ to the right or to the left according to Table 3.1. Finally, the translated beams are merged to produce the new transferred beam profile.

To provide a visual understanding of the above algorithm, we show an example in

Fig. 3.10. The figure shows the three steps of the transfer algorithm. First, the original profile is divided into two segments, each of which contains a peak. Second, both segments are shifted with $\Delta\theta_{P_1 \to P_x} = 16°$. The shifting direction is based on the orientation of $O_{P_1 \to P_x}$ and the beam category. In this example, the two RF beams are shifted to the right and to the left, respectively. Finally, both shifted segments are merged into a single one as the transferred profile. The disconnection because of the shifting might be filled by linear interpolation. The bottom figure compares the original profile (red), the transferred profile (yellow), and the ground truth (blue, acquired by FBB). The transferred profile well matches the ground truth regarding the two major beams.

3.5.3 Transfer from Multiple Profiles

Each unknown tag is surrounded by multiple reference tags. Thus, we can actually obtain multiple beam profiles for reference. The question now is how they can be fused to create the final transferred profile. As mentioned earlier, the similarity of the profile is related to not only the direction difference but also the distance. We choose the K nearest reference tags close to the unknown position P_x for the transfer. Let $\{\mathbf{B}(P_1), \ldots, \mathbf{B}(P_K)\}$ denote the beam profiles acquired from the K reference tags located at $\{P_1, \ldots, P_K\}$. Using the above transfer algorithm, we can obtain K transferred profiles denoted by $\{\mathbf{B}(P_1 \Rightarrow P_x), \ldots, \mathbf{B}(P_K \Rightarrow P_x)\}$, which are used to light up P_x. The final transferred profile is fused by the weighted mean of these K transferred profiles as follows:

$$\mathbf{B}(P_x) = \frac{1}{\sum_{k=1}^{K} \frac{1}{d_{P_k \to P_x}}} \sum_{k=1}^{K} w_k \mathbf{B}(P_k \Rightarrow P_x) \text{ and } w_k = \frac{1}{d_{P_k \to P_x}} \tag{3.14}$$

where w_k is the normalized weight related to the distance between P_x and P_k, thus enabling the transferred profiles from closer positions to exert more impact on the fused profile.

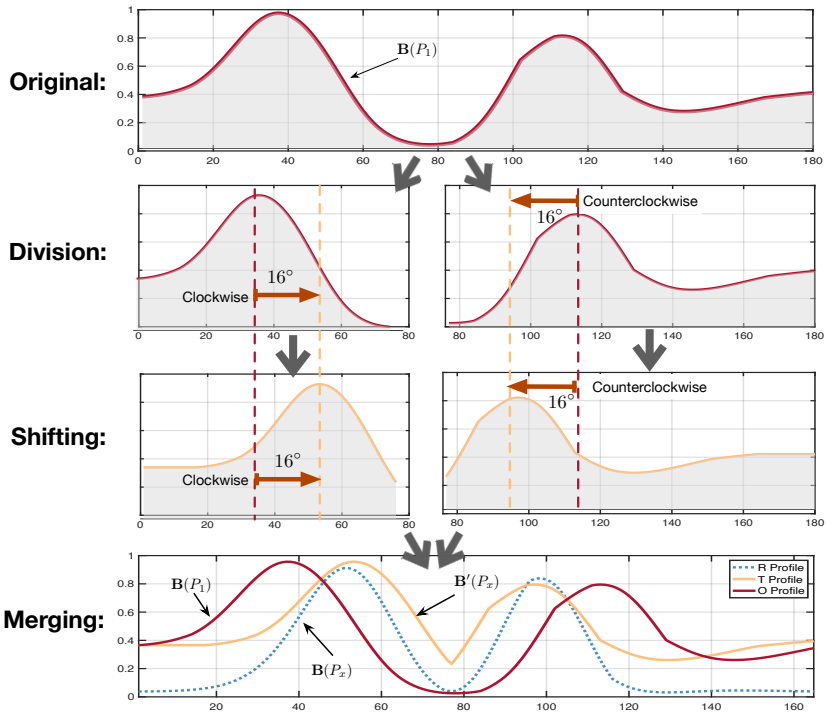


Figure 3.10: Transferring beam profile for P_1 to light up P_x where the direction difference of two positions is 16°. The process takes a total of three steps: the division, the shifting and the merging. The bottom figure compares the original (O), the transferred (T) and the ground profiles (R).

3.6 Re-forming Beams For Unknown Tags

Finally, we elaborate on the approach of reforming the transferred beams to power up the tags at unknown positions.

3.6.1 Reversing Strategy from Beam Profile

After obtaining the transferred beam profile, we need to reverse the strategy from the profile and apply it to the antenna array to form the real RF beams. Given a beam profile, how can we determine a beamforming strategy? We know that $f(\mathbf{\Phi}(P_x)) = \mathbf{B}(P_x)$ is a deterministic function and the value can be calculated using Eq. 3.9.

Unfortunately, the reverse function $f^{-1}(\mathbf{B}(P_x)) = \mathbf{\Phi}(P_x)$ is an uncertain function in that many different strategies might lead to same beam profile. Given a beam profile $\mathbf{B}(P_x)$, our goal is to address the following optimizayopm problem:

$$\mathbf{\Phi}(P_x) = \underset{\mathbf{\Phi}}{\operatorname{argmin}}\ \mathcal{L}_2(f(\mathbf{\Phi}(P_x)) - \mathbf{B}(P_x)) \tag{3.15}$$

where $\mathcal{L}_2$ is the L2-norm and the dimension of $\mathbf{B}(P_x)$ is far higher than that of $\mathbf{\Phi}(x)$. A straightforward approach is to build a mapping table between them in advance by visiting all possible strategies and generating the corresponding beam profiles. The appropriate strategy can be found by comparing the given profile with all indexed profiles. This approach is only a preliminary solution for scenarios where we have powerful server or are not sensitive to the latency. This is because it consumes a large amount of disk for storage space and takes a long query time due to the numerous possible strategies as mentioned earlier. However, it can not work especially in dynamic environments that demand fast updates or scenarios where computation resources are critical considerations.

With the fast development of deep neural network (DNN) recently, DNN has demonstrated great power in solving an optimization problem like ours. Thus, we build a simple fully connected neural network (FCN) to fit the reverse function $f^{-1}(\cdot)$. The network takes a beam profile $\mathbf{B}$ as input and outputs beamforming strategy $\mathbf{\Phi}$, with the aim of minimizing the above loss function. The FCN consists of an input layer, five hidden layers, and one output layer. The input layer has 180 neurons, which can accept a beam profile in which the angle varies from 0° to 180°. The following hidden layers contain 256, 512, 1024, 512, and 256 neurons, respectively. The output layer contains 8 neurons because there are 8 transmitting antennas. The loss function can be formulated as $\mathcal{L}_2(f(\mathbf{\Phi}) - \mathbf{B})$. This neural network is independent of the signal propagation model and the real environment settings but purely aims to resolve a mathematical optimization problem. Thus, we can randomly generate a large number of strategies and their corresponding beam profiles to train the network. We use stochastic gradient descent (SGD) to train the model.

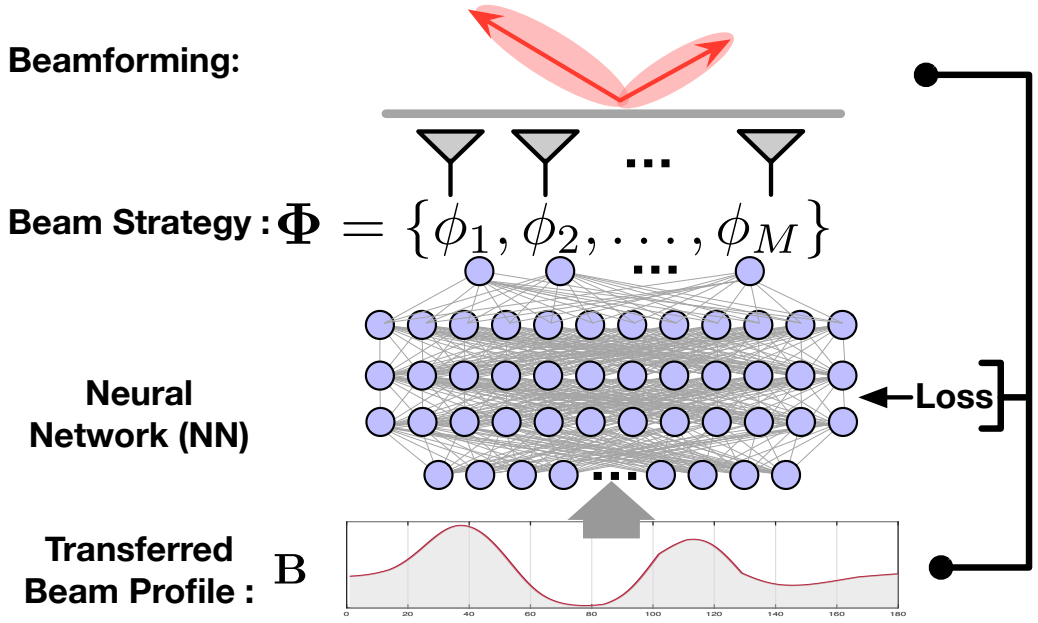


Figure 3.11: Finding a strategy using the FCN. A FCN is used to find the corresponding strategy for a given transferred beam profile. The real RF beams are formed and transmitted after the beamformers are configured based on the output strategy.

As shown in Fig. 3.11, we will use the FCN to find the beamforming strategy after it is fully trained. Specifically, given a transferred beam profile, it is fed into the FCN and a strategy is output. With the application of the phase configurations based on the output strategy into transmitting antenna, the antenna array will form the real RF beams in the desired fashion.

3.6.2 Powering Up All Unknown Tags

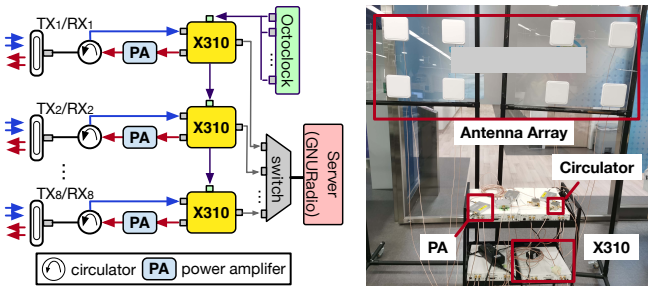
So far, we can determine a beamforming strategy for a given position. However, we do not know where the unknown tags of interest are located. To address this issue, we divide a container (e.g., a shelf) into many grids (or cubes), each of which is $\lambda/2 \times \lambda/2$ in size, i.e., a half-wavelength is chosen to ensure that constructive interference can be always achieved within the grid. This size is the smallest area that our beamforming

can identify. All unknown tags will be powered up after all the potential grids are lit up. To speed up the traversing time, we will broadcast a `Query` command including one time slot in the beginning, such that all tags must reply the first time. Only when any backscatter signal is detected in the slot, can the anti-collision procedure be launched. Otherwise, we visit the next grid directly.

Optimization. The range of a commercial reader is about 15 m. Regarding the directivity of the antenna, the array is supposed to cover a half circle of $2\pi * 15^2/2 = 706.5$ m^2. In theory, it seems that we still have $706.5/0.16^2 = 27,597$ grids to visit where $\lambda = 32$ cm. Actually, an RF beam can power up all tags deployed along a propagation path. Thus, one strategy covers multiple grids simultaneously instead of a single grid. Many redundant strategies must exist. Before really forming the RF beams, we can find out all strategies and compress the redundant strategies, which have the same number of RF beams and these RF beams direct at the same directions. Our practical experience suggests that the compression ratio is up to 12%-that is, about 88% strategies are redundant. This upper bound estimation is derived from our field study in a warehouse and the comprehensive analysis can be found in § 3.8.4. Surely, this ratio may vary depending on the practical layout and specific conditions of different applications. As a result, only about $27,597 * 0.12 = 3,311$ strategies are left and it takes about $3,311 * 0.01 = 33$ seconds to scan the whole region. The ratio highly depends on the practical layout. Further optimization can be performed based on temporal or spatial patterns. For example, the items on some shelves will always be frequently updated; some shelves are always empty in some regular periods, and so on. However, this topic is beyond the focus on transfer beamforming in this work and will be studied in the future.

3.7 Implementation

We introduce the prototype implementation of TBF.



(a) Hardware Architecture (b) Prototype

Figure 3.12: Hardware implementation of our custom-built reader.

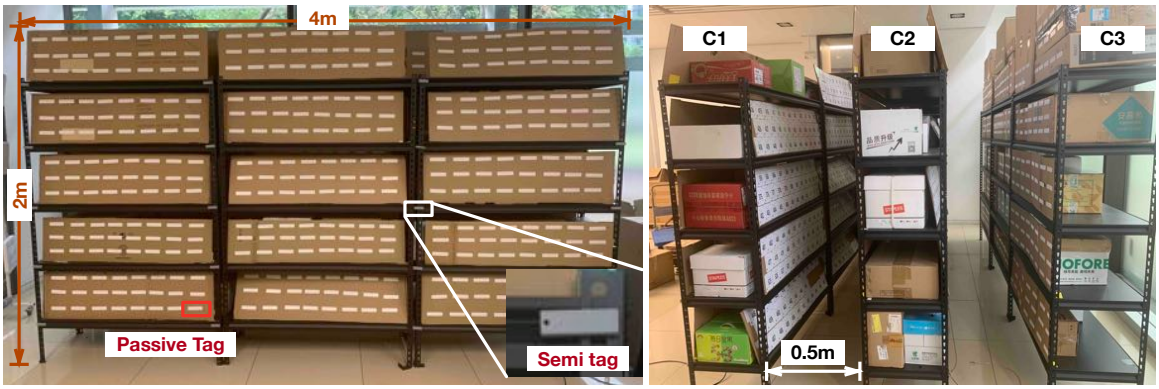


Figure 3.13: Testing environment setup. We attach 2,160 commercial RFID tags in a dense configuration to boxes and deploy them on metal shelves.

(a) TBF hardware. The block diagram in Fig. 3.12(a) details the hardware architecture of TBF's customized RFID reader, and the prototype reader is shown in Fig. 3.12(b). We use 8 commercial UHF RFID antennas to form an antenna array. Each antenna has 5 dBi gain and 120° beamwidth. We use a circulator to separate the TX and RX signals for each antenna. We use four USRP X310 units to emulate the eight duplex RF chains [14], where each RF chain is connected to an antenna. Each X310 radio is equipped with two Ettus SBX daughterboards [13] that allow for communication on the 902-928 MHz UHF RFID band. All X310 radios are time-synchronized utilizing an OctoClock CDA-2990 [11]. For each TX chain, we use a SKY65162-70LF power amplifier to amplify the TX signal to 25 dBm [102]. The

overall output power of the array is 2 W under the FCC RFID regulation [39]. We use a signal source to calibrate the initial phase offset of each RF chain. We use a 10 Gb switch to connect all USRP X310 radios to a Linux server with 12 cores and 64 GB RAM. The sampling rate for each RF chain is 2 MS/s.

(b) TBF software. The core beamforming algorithm and RFID signal decoding are implemented in GNURadio v3.9.0 and UHD v4.0.0 by significantly extending the code in [58]. We develop a digital phase shifter to adjust the baseband phase of each TX chain. In the RX chain, we measure the tag's uplink phase by using the algorithm in [79], which can effectively solve the π-ambiguous problem. The neural network for generating the beam strategies is developed by using the PyTorch framework [12]. We collect 8,400 data samples in the dataset and divide them using an 80/20 training/test split. The model is trained with an AMD 5900x (4.9GHz) processor, 64GB RAM, and 2 NVIDIA 3080Ti GPUs. Stochastic gradient descent is used to optimize the training process with a learning rate of $1 \times e^{-3}$ and momentum of 0.9 across the whole experiments. The batch size is set as 128. Training the model takes about 4 hours.

(c) RFID tags. We use two types of UHF RFID tags. The semi-active tag is an EM Microelectronic EM4325 [78] working in battery-assisted mode. Its reading sensitivity is -31 dBm, which is 16 dB better than that of the passive tags. The passive tag is the Alien 9654 [15]. An EM4325 tag costs 3 USD, and an Alien tag costs 0.05 USD. We deploy a total of 150 semi-active tags and 2,160 passive tags in our evaluation.

The code and the dataset have been released on GitHub: `https://github.com/yxy1995123/Transfer-Beamforming-via-Beamforming-for-Transfer-TBF-`

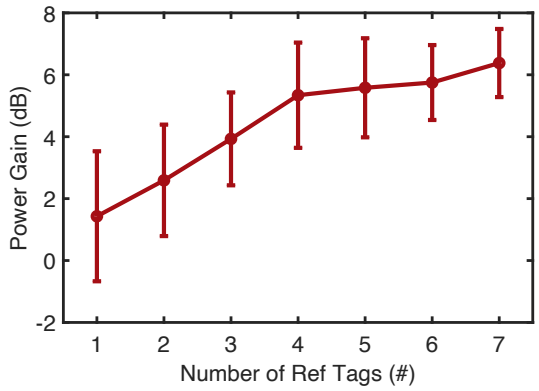


Figure 3.14: Impact of # of reference tags

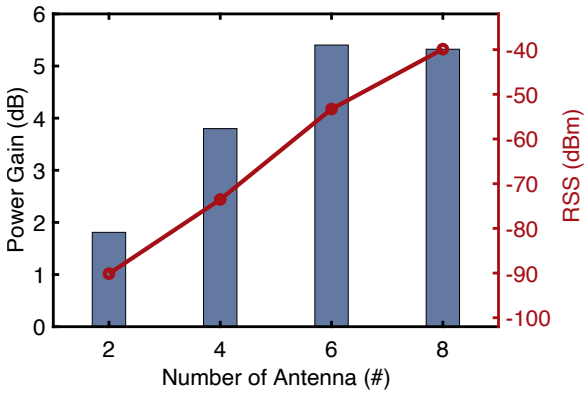


Figure 3.15: Impact of # of antenna.

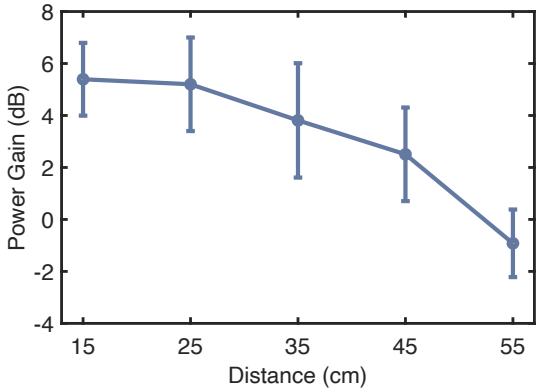


Figure 3.16: Impact of distance

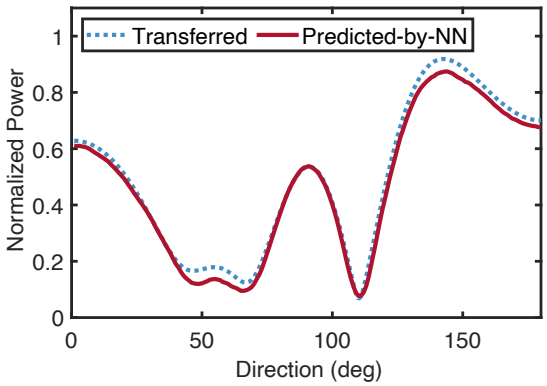


Figure 3.17: Beam profile generation.

3.8 Results

3.8.1 Setup

We construct a testing scenario to simulate the real-world warehouse operation, as shown in Fig. 3.13. We place a total of $9 \times 5 \times 2 \times 24 = 2{,}160$ commercial passive tags in a dense setup. Twenty-four passive UHF RFID tags are attached to cardboard with dimensions of 1.2 m (L) $\times$ 0.4 m (H). The distance between two adjacent tags is 5 cm. All cardboards are attached on the 9 metal shelves with dimensions of 1.3 m (L) $\times$ 0.6 m (W) $\times$ 2.2 m (H) and 5 layers. Both the front and back of each layer are covered with cardboard. Nine shelves are arranged in three rows, with three shelves

in each row. We place printouts or books in the middle of the two cardboards. The reference tags are evenly attached to the shelves. The antenna array is deployed at the side of the warehouse and is 3 m away from the shelf.

Baseline. We compare TBF against the state-of-the-art blind beamforming algorithm: PushID [116], which is a distributed beamforming system. There are few other open source beamforming systems that can be accessed. While PushID's primary focus may be on range enhancement, it still provides a valuable basis for evaluating the coverage with beamforming technology based on the classic geometry model. This alignment with our evaluation criteria makes it a suitable baseline for our study. In our scenarios, the antennas for the PushID system are evenly located along the warehouse wall and kept 3 m away from the shelf.

Deployment Overhead. Semi-active tags (used as reference tags) are more expensive than passive tags, their deployment is strategic and limited in number which may limit the application in large-scale system. In our proposed solution, we require only 150 active tags, costing 3 USD each, amounting to 450 USD. When considering the overall system cost, including the reader and the enhanced performance and coverage provided by the TBF approach, the additional expense is justified. The total cost of 3,750 USD is still only 31% of the cost of using non-beamforming readers, as demonstrated in our warehouse scenarios. The deployment of reference tags is indeed an additional step, but it's a one-time effort that brings substantial long-term benefits. Considering the huge benefits of reference tags, we foresee that in the future, warehouse equipment manufacturers may embed reference tags directly into shelves or bins. This integration would further reduce deployment costs and improve the inventory efficiency of RFID systems, making our approach even more appealing. In response to this concern, we have included a comprehensive discussion on the cost considerations and deployment overhead of the reference tags.

3.8.2 Power Gain

The goal of TBF's beamformer is to boost the power transmission to backscatter nodes in complex wireless environments to solve the misreading problem. To evaluate the effectiveness of TBF, we compare the transmitted power of TBF with that of the baseline. We measure the signal power by using a dedicated USRP receiver with an RFX900 antenna connected to the receiver with a shielded RF wire. We compute the power gain as the square of the ratio between the carrier signal strength with TBF and baseline. Unless specified otherwise, all error bars in graphs denote standard deviation.

■ **Impact of reference tag amount.** We first evaluate how reference tags help the power transmission and how it is impacted by the number of reference tags K. We use 8 antennas and measure the power gain at 100 different positions on the shelves. For each position, we place K reference tags in a ring around it at a distance of 15 cm. Fig. 3.14 shows the power gain of TBF compared with that of the PushID algorithm. We make the following observations: (1) The power gain monotonically increases as K increases from 1 to 7 and achieves a 6.9 dB gain when $K = 7$. This finding shows that more reference tags can help better model the wireless channel under complex environments and improve the beamforming performance. (2) The benefit from an increased reference tag effort has a marginal effect. When $K = 4$, we have a maximum per-tag gain, and the overall gain is 5.34 dB. Four reference tags are recommended for most cases. Unless specified otherwise, we set $K = 4$ in the following experiments. (3) Using only one reference tag can not ensure positive power gain in all cases. This finding shows that a single beam profile may mislead the beamformer in some complex scenarios.

■ **Impact of antenna number.** Next, we would like to understand the impact of the number of antennas. We fix the number of reference tags $K = 4$ but vary the number of transmitting antennas and repeat the above experiments. Fig. 3.15

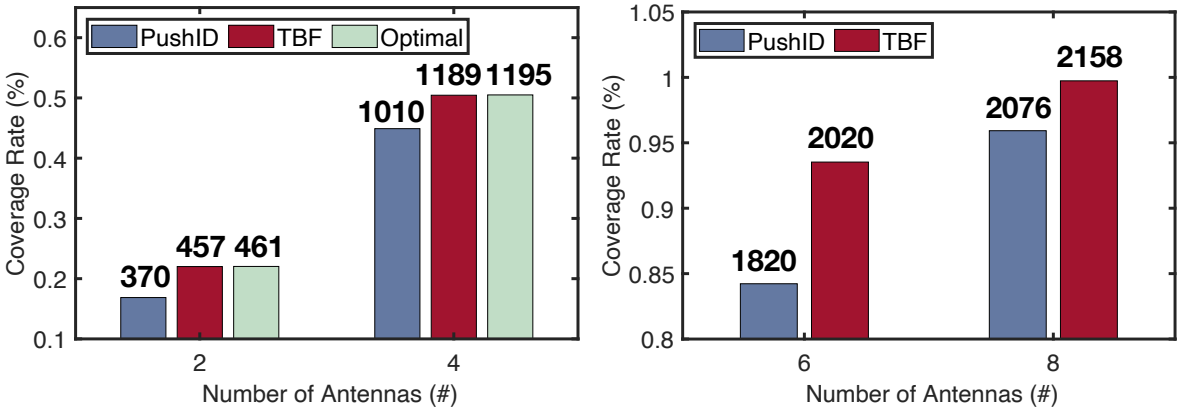
plots the median power gain compared with PushID and the median received signal strength (RSS) of the reference tags. The figure shows that TBF has higher power gain when more transmitting antennas are used. When 2 transmitting antennas are used, the power gain is about 1.2 dB. The trends of the RSS of the reference tags and the power gain are relatively similar. Hence, we speculate that the weak signals of reference tags increase the error of channel estimation and beam profile when fewer antennas are used.

■ **Impact of reference tag distance.** We then evaluate how the spatial interval between the reference tag and desired position impacts the power gain. We still use 4 reference tags to transfer the beam but vary the distance between the reference tags and the desired ones. Fig. 3.16 shows the results. As the distance increases from 15 cm to 55 cm, the power gain drops from 4.2 dB to -2 dB. This result is expected because the closer tags have more relative beam profiles and better transferring gain. When the interval is larger than 45 cm ($\approx 1.5\lambda$), the power gain of TBF is worse than that of the past model-based method, which shows that the wireless channels of reference tags differ significantly from that of the transferred tags.

3.8.3 Coverage

In this experiment, we evaluate how TBF improves the inventory coverage compared with the baseline in the testing warehouse. The evaluation is based on two performance metrics: (1) **Coverage rate**, which is the fraction of tags that are inventoried over the total 2,160 tags. To reduce the effect of the randomness of the RFID Aloha protocol on the results, we repeated the inventory 20 times and took the average. (2) **RSS** is the signal strength of received backscatter signals from activated tags. RSS further indicates the wireless link quality. We measure the RSS of the tags' signal by using a USRP receiver with a single antenna.

■ **Coverage rate.** To validate the effectiveness of TBF, we compare the TBF with



(a) 2 and 4 antennas

(b) 6 and 8 antennas

Figure 3.18: Coverage rate. The number above the bar is the number of tags inventoried.

PushID and optimal beamforming strategies. We perform an exhaustive search to find the optimal beamforming strategies when the number of antennas is 2 and 4. We omit the optimal results for 6 and 8 antennas because the search space is too large. Fig. 3.18 shows the percentage of tags discovered when different numbers of transmitting antennas are used. We make the following observations: (1) TBF outperforms the baseline across all four settings. The coverage rates of TBF are 21.2%, 55.5%, 93.5%, and 99.9% using 2, 4, 6 and 8 transmitting antennas while those of PushID are 17.1%, 46.8%, 84.3%, and 95.8%, respectively. The average coverage rate of TBF is 6% higher than the baseline. TBF can find 82 and 100 more tags than the baseline when 6 and 8 antennas are used, respectively. Such improvement is crucial to the application of modern large-scale logistic networks. (2) TBF can achieve a 99.9% coverage rate when 8 antennas are used in our testbed. Only 2 of 2,160 tags are missed. Such coverage performance can meet the demand of real-world logistic networks [129]. (3) When 2 and 4 antennas are used, TBF can discover 457 and 1,189 tags, respectively. We note that optimal strategies can find only 4 and 6 more tags. This result indicates that the performance of TBF is close to the optimal.

■ **RSS gain.** Next, we investigate the RSS of the inventoried tags in line-of-sight

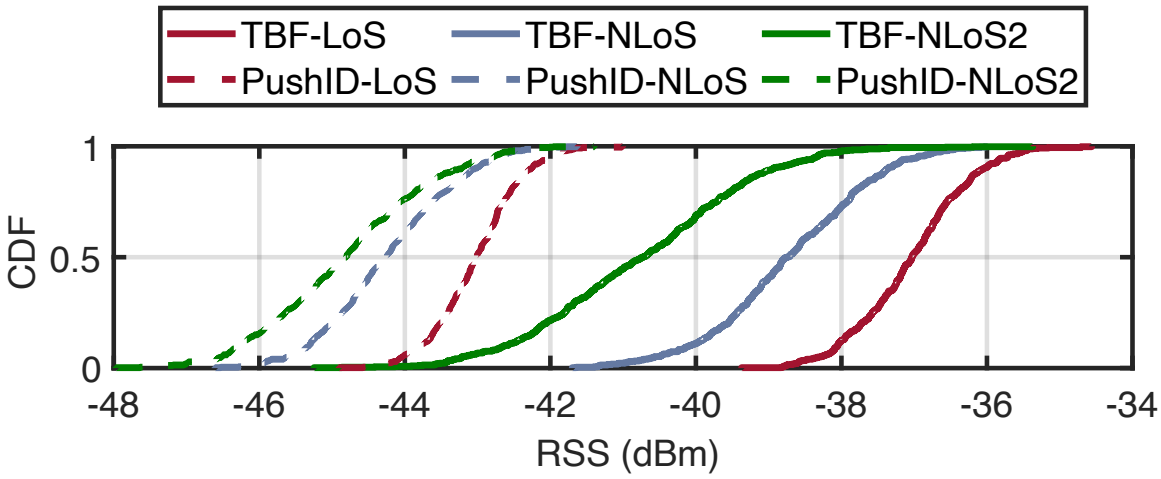


Figure 3.19: RSS distribution.

(LoS) and non-line-of-sight (NLoS) settings. Two NLoS scenarios refer to the second and third rows of shelves, which are totally blocked by the first-row shelves. Fig. 3.19 plots the RSS distribution of the collected backscatter signals of TBF and PushID when 8 antennas are used. The median RSS of TBF for three scenarios are -36.92, -38.73, and -40.78 dBm, while those of PushID are -43.04, -44.31, and -44.78 dBm. On average, TBF improves the RSS by 6.1 dB for the LoS scenario and 4.78 dB for the NLoS scenario. Such RSS gain results are in line with the power gain experiments. This result shows that the model of TBF works well in both LoS and NLoS settings.

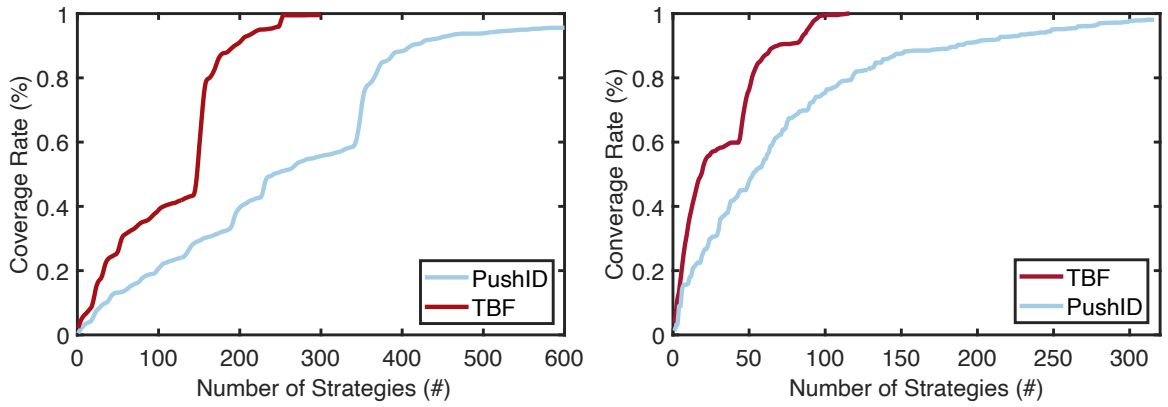
3.8.4 Convergence

Next, we analyze the inventory convergence time of TBF and the compressed ratio of beam strategies.

■ **Convergence.** Both TBF and PushID can find the set of beamforming strategies offline. The main computational bottleneck of the inventory is the rate at which beamforming strategies are applied to the hardware, which is about 100 ms in our prototype. Hence, we measure the inventory speed in terms of the number of beam-

forming strategies that need to be applied. Specifically, we setup two scenarios, one is the complex warehouse application where 2160 tags are densely stacked on the front and back of the shelves as shown in Fig. 3.13 and another is a typical supermarket application where 1536 tags are only placed on the outsides of the shelves. Fig. 3.20 shows the percentage of tags discovered by two methods in two cases as we increase the beamforming strategies. We make the following observations: (1) The inventory curve of TBF increases faster than that of PushID in both two cases. Taking the warehouse case as an example, TBF uses 147 and 195 beamforming strategies to cover 50% and 90% tags, respectively, while PushID requires 241 and 412 strategies to achieve the same coverage. TBF saves time by 39% and 52% for 50% and 90% coverage, respectively. This result occurred mainly because TBF can find more effective beamforming strategies with the help of the reference tags. (2) In warehouse case, TBF finds 2,158 tags (i.e., 99.9% tags) by using 300 beamforming strategies, while more than one-third of strategies (i.e., 105 strategies) are used to find the last 5% tags (i.e., 108 strategies). On average, one strategy can find only one unique tag. This result shows the complexity of wireless environments and the difficulty of modeling the wireless channel. (3) In supermarket setting, TBF achieves 50% and 90% coverage rate by using 21 and 84 beamforming strategies, while PushID needs 54 and 189 strategies to achieve the same coverage. This result indicates that even if the area of interest removes a portion of the NLOS cases, the accuracy classic channel modeling method for reflectors is still limited and can be further improved by using TBF.

■ **Strategy compression.** The compression ratio of beam strategies is the key to beamforming efficiency. We investigate how the beam compression ratio varies when the coverage area expands. Fig. 3.21 shows the beam compression ratio for covering one shelf to nine shelves. The compression ratios for the first three shelves are very low at around 6% because they are in the LoS area and the channels are relatively simple. The ratio fluctuates around 9.5% when an area of more than four shelves is



(a) Simulated Warehouse

(b) Simulated Supermarket

Figure 3.20: Coverage rate for two typical scenarios. (a) is tested in a simulated warehouse where tags are densely stacked on the front and back of the shelves (b) is tested in a simulated supermarket where tags are only placed at the outsides of the shelves

covered. The ratio comes to 11.6% to cover the whole area. In this case, most tags are in the NLoS area with complex multipath effects, and less overlap occurs between beam strategies. We believe that most daily scenarios would be simpler than our demo scenario, making it a reasonable upper bound estimation.

3.8.5 Beam Profile Generation

We evaluate how well the neural network model extracts the beamforming strategy from the given beam profile in terms of accuracy and time.

■ **Prediction Accuracy**: Fig. 3.17 shows a sample comparison between the original transferred beam profile and that generated by the beamforming strategy which is predicted by our neural network. Clearly, the predicted beam profile well recovers the shape of the original one. In our testing, the average beam profile error calculated by Eq. 3.15 is only 0.043.

■ **Time Cost**: Next, we evaluate how deep learning based prediction method speed

up the beamforming strategy prediction compared with the basic table-looking solution. We consider the array with 8 antennas and 6-level phase shifts, there will be a total of $8^6 = 2.6 \times 10^5$ strategies. In our server settings, it will take an average of 1.2 seconds to check the whole table. With the same hardware settings, this approach only requires 0.009 seconds, making it 133 times faster than the old method.

3.8.6 Dynamic interference

In real-world applications like warehouses, there is often dynamic interference, such as moving people and robots. In theory, our system can sense the real-time conditions of the environment by continuously monitoring the channels of these reference tags and can adjust beamforming strategies to mitigate the effects of dynamic interference. In the experiment, we let a volunteer walk around one shelf to simulate a daily check at a speed of around 0.2m/s. We collect the beamforming strategies of reference tags around this shelf per 0.5s and then update the beamforming strategies in the next inventory. Fig. 3.22 depicts the plot of power gain of the tags on this shelf as the number of antennas increases. We have two observations: (1) the average power gain of 4 static scenarios is 4.2 dB and that of dynamic scenarios is 3.8 dB. There are few power loss in dynamic scenarios. This indicates that reference tags can help TBF adjust to dynamic scenarios. (2) As the number of antennas increases, the power gain gap between static and dynamic increases. This is mainly because the array with more antennas will form narrower beam and be more sensitive to the obstacles compared to that with fewer antennas.

3.9 Conclusion

This paper presents TBF, a novel blind beamforming system that can power up commercial backscatter nodes in a complex environment. The unique ability of TBF is

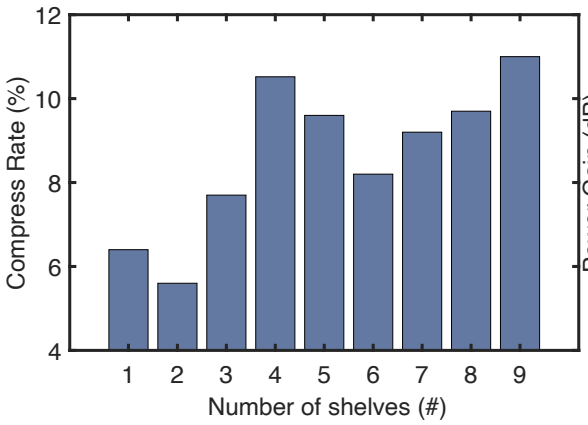


Figure 3.21: Strategy Compression.

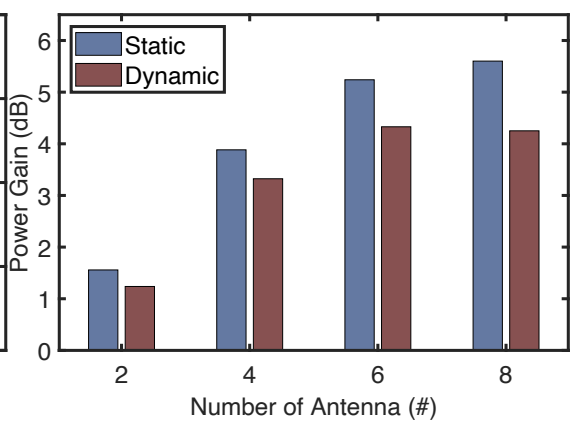


Figure 3.22: Dynamic Interference.

that it leverages the neighbor reference tags to find the optimal beamforming strategy quickly in a transferable scheme. A prototype evaluation in a warehouse with 2,160 densely deployed RFID tags reveals that TBF can improve power transmission by 6.9 dB compared with the state-of-the-art beamforming system. We believe this work will introduce a novel perspective on wireless power transmission.

3.10 Related Work

We review the related works in two aspects:

RFID systems: Recently, RFID systems have garnered significant attention from the networking community [37, 38, 127]. Much research has been conducted to increase the reading rate [17, 122], enlarge the converage [127, 38, 74, 116] and improve the reading reliability [129, 22]. The state-of-the-art work, PushID [116], pushes the RFID communication range to 64 m by proposing a distributed beamforming technology. However, it has 5% reading blind spots and can not meet the stringent accuracy demands in the real world. RFGo [22] mitigates the blind spots by using multiple antennas but works in a small, clean lane only. Unlike these methods, TBF first eliminates the persistent blind spots problem in complex scenarios via transfer

beamforming.

Beamforming algorithms: The RFID reader requires beamforming to be performed without any prior knowledge of the wireless channel. This issue is a typical blind beamforming problem. Past blind beamforming algorithms broadly fall into three main categories: opportunistic beamforming [74, 114], model-based beamforming [116, 119, 112], and heuristic beamforming [37, 38, 27]. Opportunistic beamforming algorithms such as IVN [74] perturbs the carrier signal to adapt to complex and variable channels. Model-based solutions construct the environment model by analyzing collected signals [116] or the information from other channels such as LiDar [119] and different frequencies [112]. However, both opportunistic and model-based algorithms can not ensure maximum power transmission in complex environments. Heuristic methods search for the optimal beamforming strategies by collecting receivers' feedback iteratively [38]. Unfortunately, this approach is time-consuming and can not be used for large-scale deployment. Unlike these methods, TBF explores the local similarity of the wireless channel and proposes a novel blind beamforming scheme called transfer beamforming, which can effectively improve the beamforming performance and searching efficiency.

Chapter 4

Binary Optical Neural Networks with Million-Scale Neurons

4.1 Motivation

Deep learning, a subfield of machine learning, employs artificial neural networks comprising at least three layers to approximate human cognitive processes. These networks are designed to "learn" by ingesting large volumes of training data and have shown remarkable progress in various sectors including image recognition [49], speech recognition [128], language translation [101], and medical diagnosis [20]. Presently, most deep learning architectures rely on electronic neural networks (ENNs), where neurons or layers serve as logical data structures. These ENNs are electrically powered and have recently come under scrutiny for their exorbitant energy consumption. For instance, a single V100 GPU may consume between 250 to 300 watts. Operating a GPT-3 language model, specifically MegatronLM, on a 512 V100 GPU for nine days necessitates approximately 27,648 kilowatt-hours of energy, nearly a third of an average household's annual energy use [6]. Consequently, the long-term sustainability of ENNs is increasingly questioned.

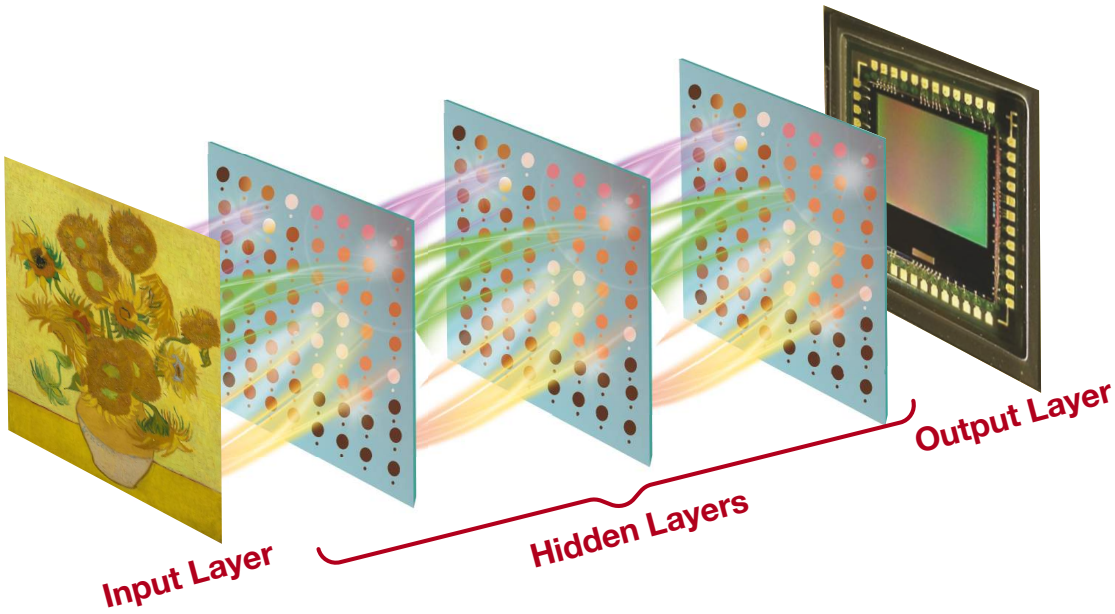


Figure 4.1: Illustration of Optical Neuron Networks. ONNs leverage the properties of optics to execute complex mathematical operations, like matrix multiplications and additions, at the speed of light. ONNs consist of physical transmissive layers, often termed optical metasurfaces. These layers house arrays of optical elements. Each optical element captures light and re-emits it, modifying its physical properties.

In the quest for sustainable alternatives, optical neural networks (ONNs) have emerged as a promising solution, drawing significant interest [68, 99, 24, 77, 118, 48, 104, 131, 80]. A representative ONN architecture is depicted in Fig. 4.1. Unlike traditional logical data structures, ONNs feature actual physical transmissive layers, commonly referred to as optical metasurfaces. Each layer consists of an ensemble of optical elements (akin to the components of an RF antenna array). These elements intercept light and re-emit it with altered physical attributes (i.e., phase, amplitude, or frequency), as a secondary source in line with the Huygens-Fresnel principle. Beyond the optical elements, the light is blocked. This entire arrangement mirrors the structure of a fully connected neural network.

- Analogous to neurons, the optical elements on metasurfaces intake optical signals from prior-layer elements and retransmit modified signals to elements on the next layer.

- Analogous to weights, the altered optical signals are modulated by multiplying them with the adaptable transmissive coefficients inherent to the optical elements.

- Analogous to connections, optical signals emitted from optical elements on one layer are linearly superimposed at the elements on the next layer because of the diffraction.

More technical details about the ONN refer to § 4.2.

ONNs leverage the inherent efficiencies of optical processes to execute complex matrix multiplications and additions at light speed. Unlike their electronic counterparts, ONNs discard energy-intensive transistors in favor of more sustainable, sometimes even energy-neutral, optical components. As optical signals carrying relevant information (e.g., images) traverse these layers, the required additions and multiplications are intrinsically executed due to the principles of linear superimposition and optical wave diffraction. ONNs circumvent the need for intermediary procedures common in ENNs such as register shifting, cache paging, and onboard communication. This eliminates the overheating challenges often induced by rapid electron movement, rendering ONNs both faster and more energy-efficient than ENNs.

However, crafting nanoscale optical metasurfaces poses notable technical challenges [36]. For optimal transmission performance, each 'neuron', or optical element, should be dimensionally commensurate with the light wavelength. The transmission coefficients of these neurons are dictated by their geometric characteristics, such as thickness and shape. Consequently, the majority of ONNs currently rely on costly electron-beam lithography (EBL) for meticulous neuron shaping, with fabrication costs escalating to as much as 10,000 USD per layer for a one cm^2 metasurface [65, 5].

Figure 4.2: BONN Prototype. The left shows a layer of BONN with an area of 0.8×0.8 mm^2 on a finger, which accommodates one million 800ns-diameter neurons. The right shows the zoomed-in layer under a 1000× microscope, where black circles are the binary neurons.

To curb these expenses, recent studies like D2NN [68, 24] have resorted to lower-frequency ONN (i.e., 400GHz) While this approach has indeed trimmed costs to a degree, it introduces two primary constraints. First, a diminutive layer consisting of a 100×100 neuron grid spans to 7×7 cm^2. When expanded to accommodate neurons on the scale of millions (for instance, 1000×1000), the required area balloons to 70×70 cm^2. Such expansive layers pose integration challenges, especially in compact devices. Second, these designs are incapable of modulating visible light, which obstructs their seamless integration into imaging devices. Consequently, there remains a pressing need for ONNs operating within the visible spectrum that are also economically viable.

In this work, we introduce the Binary Optical Neural Network (BONN), an economically viable, large-scale optical neural network featuring nanoscale neurons operational in the visible light spectrum. BONN deploys binary weights, restricting each weight to either 0 or 1. Binary neural networks (BNNs) have long been explored for minimizing computational loads in resource-constrained devices [90], demonstrating that this pared-down architecture can still maintain a high level of accuracy across a

multitude of tasks. Each neuron in BONN is simplified to a tiny, circular gate that is either activated (weight of 1) or deactivated (weight of 0). This enables the use of established fabrication techniques such as photolithography-etching (PE) or laser writing, drastically lowering the manufacturing cost to a mere 0.13 USD per layer. The size of each neuron is arond 800 nm, which facilitates the rapid scaling to 156 million neurons on a single layer within a 1 cm^2 area.

However, realizing such a nano-scale BONN encounters the following two main challenges.

• *How are the weights binarized?* The straightforward approach to develop a BNN involves employing a specific sign function that takes a real-valued weight as input and outputs a binarized pseudo-weight. This is done to maintain compatibility with the error-backward propagation training paradigm. However, the sign function presents challenges due to its nondifferentiability and imbalance. To surmount these limitations, we adopt the Gumbel-Sigmoid function as the weight function. This probabilistic function serves to push the pseudo-weights towards either of the binary extremes with an approximate 50% likelihood.

• *How is BONN trained?* ONNs consist of passive elements that are not readily adjustable during live operations. Therefore, training either a conventional ONN or a BONN mandates an offline approach rooted in simulations. Once weights are meticulously trained, the BONN is then fabricated. Consider a five-layered BONN, for instance; it demands the training of three million weights. Simulating the transit of a staggering one million optical rays between just two layers is computationally daunting. This complexity often surpasses the capacity of standard servers, posing a barrier to the widespread adoption of BONN. To address this challenge, we introduce a Fourier Optics (FO) enabled training approach. This technique transforms the intricate convolution in the spatial domain into a more manageable dot product operation in the angular spectrum, drastically trimming the simulation complexity from a quadratic scale, $\mathcal{O}(KN^2)$, to a log-linear scale, $\mathcal{O}(KN\log(N))$.

Contribution. To the best of our knowledge, this is the pioneering effort to integrate BNN into ONNs in the realm of visible wavelengths, even though both fields have been under investigation for years. Leveraging a compact design, our system is well-suited for integration into portable hardware. We successfully prototyped six prototypes of BONN within the visible spectrum, as shown in Fig. 4.2. Through exhaustive testing, we ascertained that our system delivers comparable accuracy and reliability.

4.2 ONN Fundamentals

In this section, we briefly introduce the background knowledge of ONN and discuss its characteristics.

4.2.1 System Model

Fig. 4.3 shows the top view of a typical ONN. Similar to the traditional DNN, it contains the following components:

(1) Input Layer: Parallel rays emitted from a laser source generate coherent optical signals. The input layer facilitates data modulation of the light by adjusting its amplitude, phase, or polarization. Specifically, individual pixels of an image can be encoded onto a subset of parallel rays as they traverse through the respective pixel on the planar SLM.

(2) Hidden Layers and Neurons: Prior studies have employed transmissive optical metasurfaces to serve as the layers in the network. Each of these metasurfaces accommodates $\sqrt{N} \times \sqrt{N} = N$ optical elements, also referred to as neurons, that are approximately the size of the light wavelength involved. These neurons can be meticulously shaped or etched at varying depths to calibrate their transmissive coefficients, effectively serving as the 'weights' in the network. Consequently, the physical

attributes of the optical signals that pass through are governed by these transmissive coefficients. Mathematically speaking, this corresponds to the multiplication of the incoming optical signal x by a complex-valued weight w, formulated as:

$$y = w \times x \tag{4.1}$$

Here, y is the resultant optical signal, and both x and y are complex-valued quantities.

(3) Optical Signals and Weights. Let us denote the vector of optical signals entering the neurons at the k^{th} layer as $\mathbf{X}^k$. This can be mathematically represented as:

$$\mathbf{X}^k = [x_1^k, x_2^k, \cdots, x_N^k]^T \tag{4.2}$$

where x_i^k represents the input of the neuron at the i^{th} position on the k^{th} layer and is defined by $x_i^k = a_i^k e^{\mathbf{J}\theta_i^k}$. The signal is a complex number that consists of the initial amplitude a_i^k and initial phase θ_i^k. The term $\mathbf{J} = \sqrt{-1}$, signifying a complex number. Further, the set of learnable weights for the neurons on the k^{th} layer as $\mathbf{W}^k$. It is expressed as:

$$\mathbf{W}^k = [w_1^k, w_2^k, \cdots, w_N^k]^T \tag{4.3}$$

where w_i^k refers to the learnable weight of the neuron at the i^{th} position on the k^{th} layer. It is defined as $w_i^k = \Delta a_i^k e^{\mathbf{J}\Delta\theta_i^k}$. Δa_i^k and $\Delta\theta_i^k$ represent the changes in amplitude and phase, respectively, resulting from the adjustments to the corresponding optical element. The output optical signal from a neuron is determined by the formula $y_i^k = w_i^k x_i^k$. Once the signals have traversed the k^{th} layer, the resulting signals from that layer can be described as:

$$\mathbf{Y}^k = [y_1^k, y_2^k, \cdots, y_N^k]^T = \mathbf{W}^k \odot \mathbf{X}^k \tag{4.4}$$

where $\odot$ stands for the Hadamard product operation. Typically, ONNs from prior studies utilize weights that only vary in phase, which implies $|w_i^k| = 1$.

(4) Bias. In accordance with the HuygensFresnel principle, each neuron can be viewed as a secondary source of light, as depicted in Fig. 4.4. An associated optical

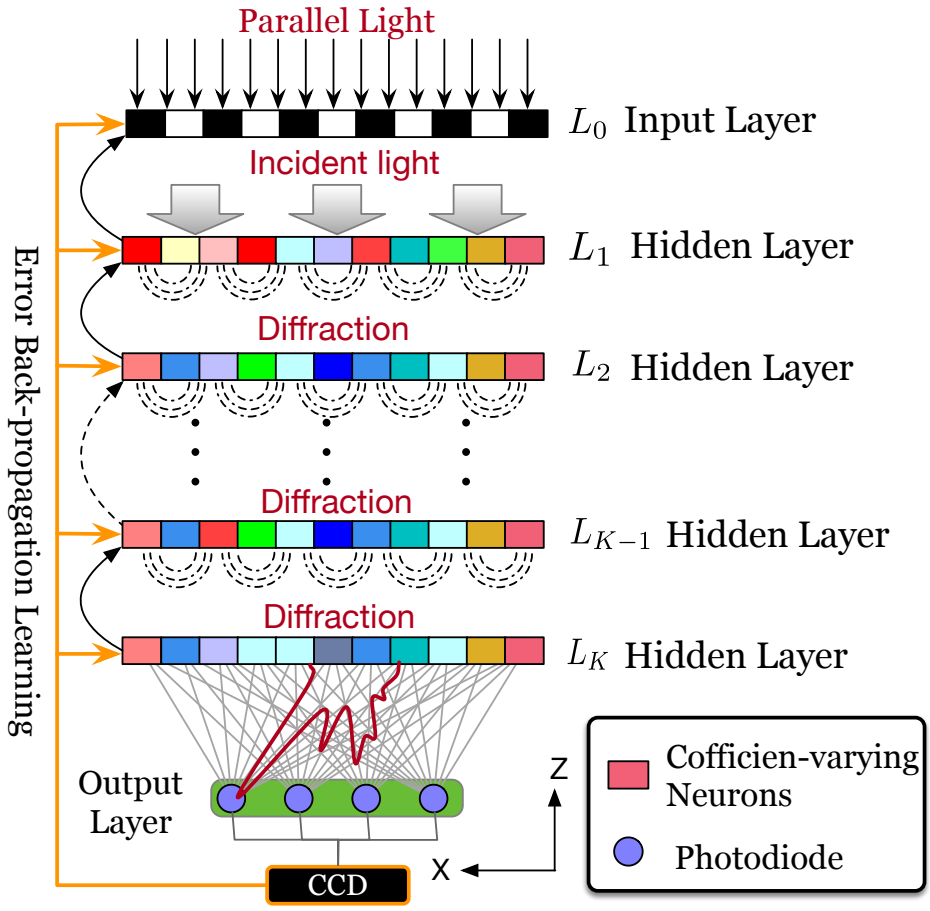


Figure 4.3: Architecture of Optical Neural Network

mode as described by Lin et al. [68]:

$$b_{P_0 \to P_1} = \frac{z}{r^2}\left(\frac{1}{2\pi r} + \frac{1}{\mathbf{J}\lambda}\right) e^{\mathbf{J}\frac{2\pi r}{\lambda}} \tag{4.5}$$

Here, P_0 signifies the source's position while P_1 designates an arbitrary point in the succeeding layer. The term λ denotes the wavelength. The symbol $r = |P_1 - P_0|$ represents the Euclidean distance between these two points. Meanwhile, z refers to the spacing between two parallel layers along the Z-axis. These emanating secondary waves from neurons in one layer converge at neurons in the subsequent layer, where they undergo interference to form the input signal. Consequently, the foundational

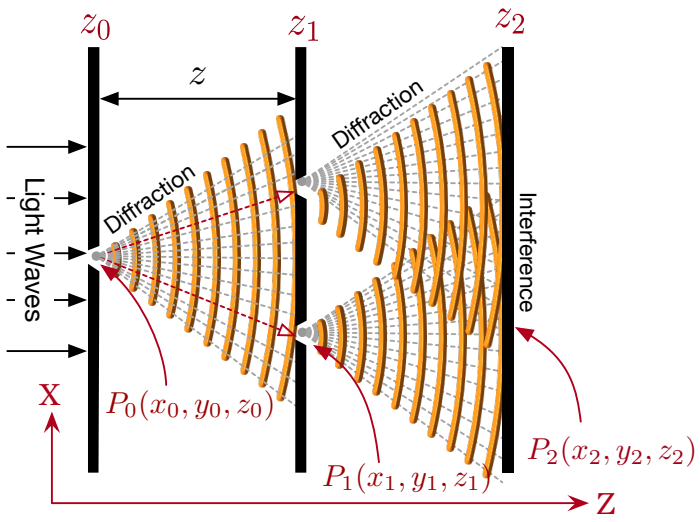


Figure 4.4: Diffraction phenomenon. Once the hole physically approximates the size of the wavelength, the light will be diffracted. The holes become secondary sources in accord with the Huygens-Fresnel principle.

matrix is constructed as:

$$\mathbf{B}^k = \begin{bmatrix} b^k_{1\to1} & b^k_{2\to1} & \cdots & b^k_{N\to1} \\ b^k_{1\to2} & b^k_{2\to2} & \cdots & b^k_{N\to2} \\ \vdots & \vdots & \vdots & \vdots \\ b^k_{1\to N} & b^k_{2\to N} & \cdots & b^k_{N\to N} \end{bmatrix} \tag{4.6}$$

where $b^k_{i\to j}$ symbolizes the bias stemming from the i^{th} neuron in the k^{th} layer directed to the j^{th} neuron in the $(k+1)^{\text{th}}$ layer. The aggregate input signal reaching the $(k+1)^{th}$ layer is defined as:

$$\mathbf{X}^{k+1} = \mathbf{B}^k\mathbf{Y}^k = \mathbf{B}^k(\mathbf{W}^k \odot \mathbf{X}^k) \tag{4.7}$$

In similar terms, the optical signal received by neuron j in layer $(k+1)$ can be represented as:

$$x^{k+1}_j = \sum_{i=1}^{N} b^k_{i\to j} w^k_i x^k_i \tag{4.8}$$

We assume the numbers of neurons on layers are identical.

(5) Activation Function. Just like traditional neural networks, ONNs also employ

activation functions to introduce non-linearity into the system. However, instead of mathematical functions, these are achieved using optical nonlinearities such as saturable absorption, optical bistability, and Kerr nonlinearity [68]. This allows the ONN to learn and model complex patterns. Consequently, the previously mentioned recursive equation (as per Eqn. 4.7) can be reformulated as:

$$\mathbf{X}^{k+1} = \mathbf{B}^k(\mathbf{F}(\mathbf{W}^k \circ \mathbf{X}^k)) \tag{4.9}$$

where $\mathbf{F}(\cdot)$ denotes the element-wise activation function.

(6) Output Layer. The output layer is the pixel array of a CMOS camera, which transitions optical signals to their electronic counterparts. The image it captures depicts the distribution of the light field. Interpretation of inference outcomes, such as categorization results, can be achieved using varying patterns. For instance, if the ONN aims to classify the input image into one of the M potential categories, the output image is conceptually partitioned into M non-overlapping sections. Each section is indicative of a specific category. If an image is classified into the m^{th} category, the corresponding section for that category becomes illuminated. Note that the process of light collection in the output layer introduces another form of nonlinearity. When CCD sensors collect light in the output layer, they only extract signal amplitude, functioning akin to a Modulo operation which is simulated to replicate the nonlinear light collection behavior.

(7) Training. The training of ONNs depends on the simulation, which determines the appropriate weights. The loss function is defined as follows:

$$\mathcal{L} = ||\tilde{\mathbf{I}} - \mathbf{I}||_2 \tag{4.10}$$

where $\mathbf{I}$ and $\tilde{\mathbf{I}}$ are the images captured by the CCD and the ground truth, respectively. And $||\cdot||_2$ denotes the Euclidean distance between the two images. The physical ONN is printed for prediction once all weights are well trained.

4.2.2 ENN versus ONN

The recursive formulas of an ONN and an ENN are combined for the comparison as follows:

$$\text{ONN: } \begin{cases} \mathbf{X}^{k+1} = \mathbf{B}^k(\mathbf{W}^k \circ \mathbf{X}^k) \\ x_j^{k+1} = \sum_{i=1}^{N} b_{i\to j}^k w_i^k x_i^k) \end{cases} \qquad \text{ENN: } \begin{cases} \mathbf{X}^{k+1} = \mathbf{W}^k\mathbf{X}^k + \mathbf{B}^k \\ x_j^{k+1} = \sum_{i=1}^{N} (b_{i\to j}^k + w_i^k x_i^k) \end{cases}$$

Both recursive equations bear striking similarities, yet two primary distinctions exist. Firstly, in an ENN, the weights are associated with the connections between neurons, resulting in N^2 learnable weights. In contrast, ONNs have only N learnable weights, as these are linked to the neurons rather than their connections. Secondly, ENNs add biases to incoming signals (i.e., data streams), while ONNs multiply them owing to diffraction propagation properties. In ENNs, biases are learned individually. Conversely, in ONNs, biases are predominantly influenced by the Z-axis distance between consecutive layers which will be explained later.

• **Deep or Wide?** Considering a network with K layers, each housing N neurons, the total number of learnable parameters in an ENN approximates $\mathcal{O}(KN^2)$. In contrast, an ONN has around $\mathcal{O}(KN + K)$ learnable parameters. Viewed from this angle, the learning capacity of an ENN significantly outpaces that of an ONN with similar depth, unless the neuron count per layer in the ONN is elevated to N^2. This observation implies that ONNs benefit more from width than from depth; in other words, $K \ll N$. This is particularly relevant as light intensity diminishes upon traversing each layer, resulting in limited light availability at the output layer in excessively deep ONNs. Thus, ONNs are better suited to a wide and shallow network architecture.

• **Why prefer ONN?** Despite their comparatively lower learning capacity, ONNs benefit substantially from their innate ability to execute complex multiplications and additions at the speed of light. Traditional neural networks demand extensive computational resources for numerous neuron calculations. However, in ONNs, these processes occur naturally as light propagates through neuron sets, enabling instant

calculations without additional energy costs and maintaining constant light source power at roughly 5 mW. Unlike ENNs, which struggle with thermal issues from joule heating, photon-driven ONNs are largely unaffected. Thus, despite some limitations, ONNs offer significant potential due to their unique advantages.

4.3 Binary Optical Neural Network

In this section, the challenges suffered by current ONNs are first introduced, and then the design of BONN is discussed.

4.3.1 Why to Binarize Optical Neural Network

To simplify the fabrication process and lower production costs, we introduce the BONN. In this approach, when a neuron's weight is learned to be 1, it is activated; otherwise, it remains deactivated. While previous ONN designs relied on the costly and intricate EBL process to adjust neuron thickness or shapes for phase variation, our BONN employs the more direct and cost-effective PE method to create holes for neurons with a weight of one, denoted as $|w_i| = 1$. This seemingly minor modification leads to a significant breakthrough in production, reducing costs by a staggering factor of 769×, translating to a mere 0.13 USD for each layer of the BONN. Next, let us elaborate on the design of BONN.

4.3.2 Binarization

The concept behind BONN draws inspiration from the binary neural network (BNN). BNNs are specialized neural networks that utilize one-bit weight values, specifically 0 and 1. Historically, BNNs were introduced to facilitate compact neural network models suitable for resource-limited embedded devices. In contrast to traditional

neural networks, BNNs offer advantages in terms of storage, computational efficiency, and energy conservation. Prior research on BNNs [123, 33, 89, 88, 90] has evidenced that these binary models can achieve performance metrics close to state-of-the-art standards on certain tasks, such as image classification. It's important to note that the adoption of the BNN approach in designing BONN is primarily driven by an intention to alleviate the complexities and associated costs of fabricating ONNs.

(1) Sign function. In BONN, the ONN is revamped by incorporating a sign function, defined as:

$$\mathrm{S}(w_i^k) = \begin{cases} 1, & \text{if } w_i^k > 0 \\ 0, & \text{otherwise} \end{cases} \tag{4.11}$$

Consequently, the recursive formula for each neuron in BONN is given by:

$$x_j^{k+1} = \sum_{i=1}^{N} b_{i\to j}^k S(w_i^k) x_i^k \tag{4.12}$$

For the sake of clarity, we refer to w_i^k as the *true-weights* and $S(w_i^k)$ as the *pseudo-weights*. True-weights can span the entire domain without any constraints, similar to conventional neural networks. They can be optimized using gradient descent methods. The sign function serves to binarize the true-weights before they're applied to the incoming signal x_i^k. As shown in Fig. 4.5, when w_i^k is positive, then $\mathrm{S}(w_i^k) = 1$ and $S(w_i^k) \times x_i^k = x_i^k$, meaning the corresponding neuron permits the incoming light. However, if w_i^k is negative, the neuron entirely blocks light propagation because $S(w_i^k)x_i^k = 0$.

(2) Sigmoid Function. While the sign function effectively binarizes weights, its non-differentiability renders it unsuitable for BONN. This is because the non-differentiable nature of the sign function prevents the use of the gradient descent back-propagation algorithm for weight optimization. To circumvent this issue, one can employ an approximate sign function, like the Sigmoid function, which is defined as follows:

$$S(w_i^k) = \sigma(w_i^k/\tau) = \frac{e^{w_i^k/\tau}}{e^{w_i^k/\tau} + 1} \tag{4.13}$$

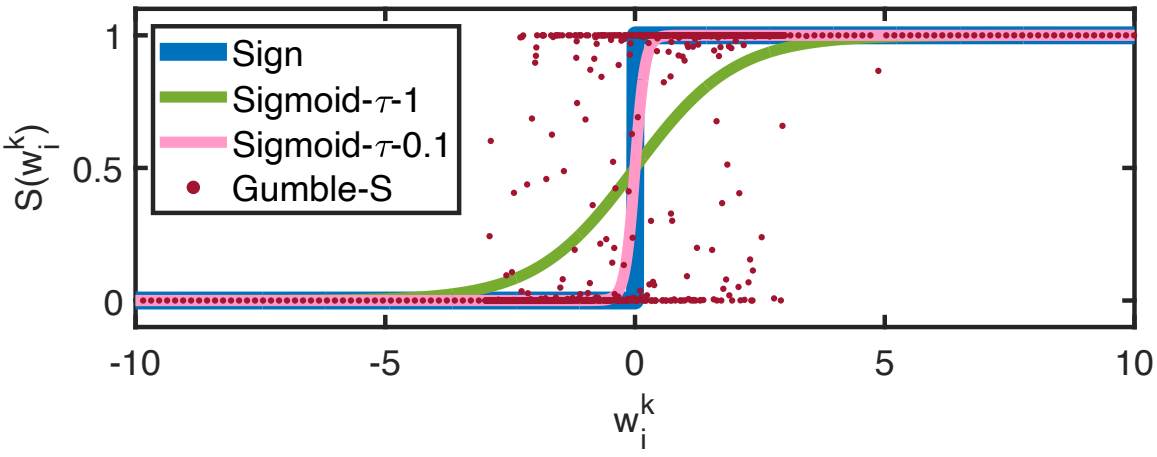


Figure 4.5: Binarization functions. They are employed to binarize the true weights learned in the entire domain into one or zero meanwhile preserving the gradient descent backward-error propagation learning approach.

where $\tau \in (0, \infty)$ is a customized parameter. The above equation represents a modified form of the Sigmoid function, traditionally applied as an activation function in neural networks. As illustrated in Fig. 4.5, when the parameter τ is diminished, the behavior of this Sigmoid function closely mimics that of the sign function.

(3) Gumbel-Sigmoid function. In BONN, neurons act as binary gates that either permits or obstructs the optical signals. Specially, when the learned weight is zero, it obstructs the light flow, effectively "closing" the gate. In contrast, a non-zero weight allows the light to propagate forward, signifying an "open" gate. Imagine a situation where the bulk of the learned weights – potentially as high as 90% – are zero. In traditional ENNs, this is less problematic. Yet, in the case of BONN, where light acts as both data and carrier, this scenario poses significant challenges. When most or all neurons are in a "closed" state, the amount of light propagating to the output layer would be little, leading to insufficient illumination on output images.

Thus, maintaining a well-balanced number of open and closed neurons is essential for effective information transmission. To help achieve this equilibrium, we introduce

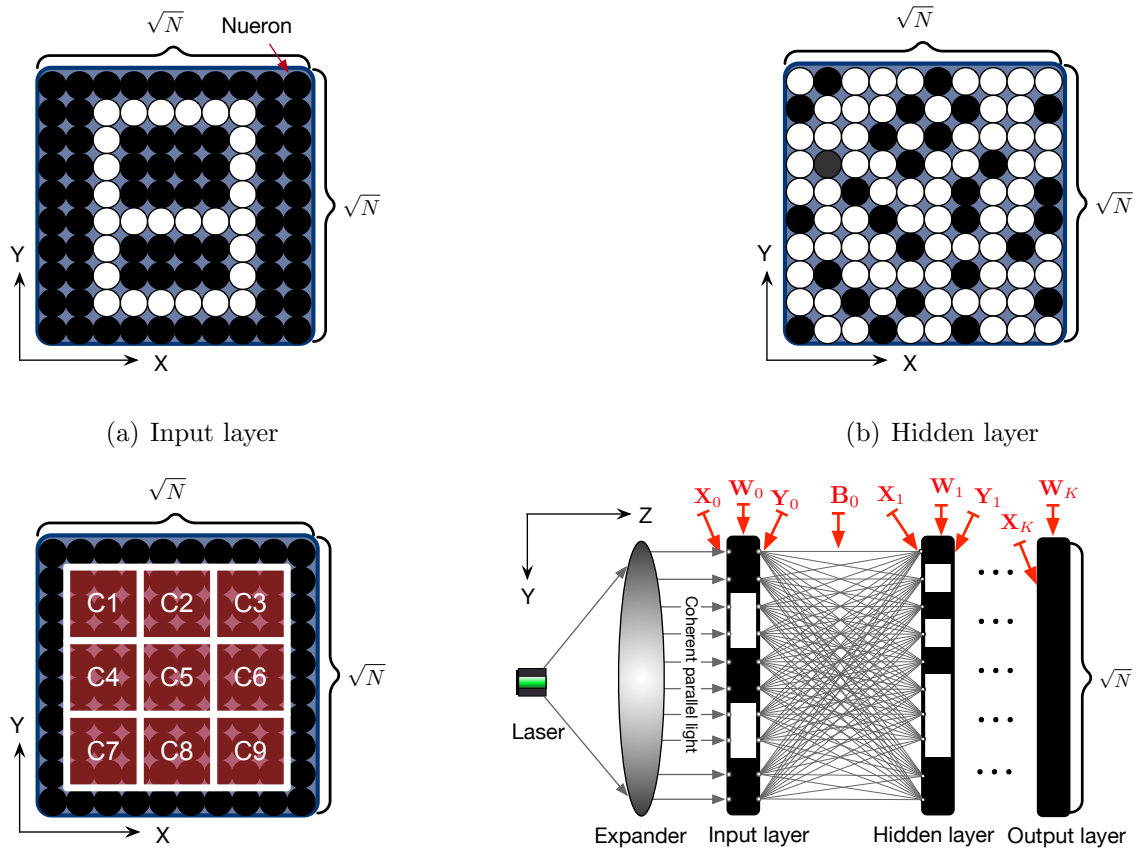


Figure 4.6: BONN neural model. (a) The physical layer is divided into continuous equal-sized holes, each of which represents a binary neuron. (d) Mathematical models for the incoming optical signals before neurons, weights, diffractive optical signals after neurons, and the propagation channels between two different-layer neurons.

Gumbel noise [47, 91, 41] into the Sigmoid function as follows:

$$S(w_i^k) = \sigma(\frac{w_i^k + g_1 - g_2}{\tau}) = \frac{e^{(w_i^k+g_1)/\tau}}{e^{(w_i^k+g_1)/\tau} + e^{(w_i^k+g_2)/\tau}} \tag{4.14}$$

where g_1 and g_2 are two independent random variables following the Gumbel distribution (i.e., density probability function: $f(x) = e^{-(x+e^{-x})}$. The above function is called the Gumbel-Sigmoid function. As illustrated in Fig. 4.5, most positive true weights gravitate towards one extreme, while the negative true weights converge towards the zero end. However, due to inherent randomness, there are instances where

positive (or negative) true weights are driven to the opposite ends. As a result of this dynamic, the proportions of 1s and 0s produced by $S(w_i^k)$ typically balance out, hovering around a 50% split. This behavior of the Gumbel-Sigmoid function ensures that a substantial portion, nearly half, of the light energy is consistently transmitted to the subsequent layer, thus circumventing potential worst-case scenarios.

4.4 Put It Together

Referring to Fig. 4.6, a consistent model represents the physical layers which encompass the input layer, several hidden layers, and the output layer. Each physical layer consists of a $\sqrt{N} \times \sqrt{N}$ grid of uniformly sized, nano-scale holes. Each hole symbolizes a neuron, which can be either open or closed. Let the binary weight $\mathbf{W}^k$ of dimension $\sqrt{N} \times \sqrt{N}$ depict the states of these N neurons for the k^{th} layer.

(1) Input layer. The initial layer, represented by $\mathbf{W}^0$, is designed to host a binarized image. An illustrative example, Fig. 4.6(a), showcases the input of a (handwritten) digit - '8'. Before feeding the image, it is resized to match the dimension $\sqrt{N} \times \sqrt{N}$ and then binarized according to their grayscale values. The processed image is then mapped onto the input layer, aligning each pixel with a neuron.

(2) Hidden layers. Spanning between the input and output layers, there exist $K-2$ hidden layers, represented by matrices $\mathbf{W}^1$ through $\mathbf{W}^{K-1}$. These layers commence with weights that are arbitrarily set but undergo iterative refinements using the backward-error propagation technique. A visual representation of one such hidden layer is provided in Fig. 4.6(b), wherein the white and black voids correspond to pseudo-weights of 1 and 0, respectively.

(3) Output layer. Concluding the series is the output layer, represented by $\mathbf{W}^K$. Neurons within this layer predominantly assume a closed status. The primary point of interest here is the light's intensity distribution, or in more technical terms, the

optical field as it interfaces with this layer. A closer look at Fig. 4.6(c) reveals the segmentation of this layer into broader sections from C_1 through to C_9, demarcating the nine distinct recognition classes. For instance, should the resulting classification align with C_1, a cluster of neurons situated within the C_1 domain becomes illuminated.

4.5 Simulation-based Training

Training ONNs involves simulation-based weight finalization before fabrication and deployment. Despite concerns over the energy used in training, it's a one-time process, while a trained ONN executes multiple times, ensuring long-term energy efficiency, especially for artificial general intelligence applications.

4.5.1 Formulization & Challenges

In Fig. 4.6(d), we illustrate the forward propagation of optical signals across layers. Light emanating from the laser undergoes transformation into a series of parallel beams by a beam expander. Upon reaching the input layer, beams that exceed neuron boundaries are obstructed. Consequently, a simulation of $\sqrt{N} \times \sqrt{N}$ parallel beams emerges, with each beam characterized by the complex value $ae^{\mathbf{J}\phi}$, where a and ϕ represent amplitude and initial phase, respectively. We employ $\mathbf{X}^k(x, y)$ to represent beams impinging on the k^{th} layer and $\mathbf{Y}^k(x, y)$ to signify beams emanating from said layer, with indices indicating neuron positions. In the following, $\mathbf{X}^k(x, y)$ and $\mathbf{Y}^k(x, y)$ are also termed *optical fields*. The objective of simulation is to ascertain the intensity of the optical field on the output layer.

The optical beams undergo either blockage or diffraction after traversing neurons. Beams that navigate through open neurons are refracted, giving rise to a new set of beams directed towards neurons in the succeeding layer due to the diffraction (see Fig. 4.4). Consequently, the light propagation between two neighboring layers is

expressed as:

$$x_j^{k+1} = \sum_{i=1}^{N} b_{i\to j}^k y_i^k = \sum_{i=1}^{N} b_{i\to j}^k S(w_i^k) x_i^k \tag{4.15}$$

This is requisite for each neuron on the $k+1$ layer.

Therefore, the computational complexity for the simulation is $\mathcal{O}(KN^2)$. Consider a case where N is set to one million, the simulation would require staggering *one tera* multiplicative operations each time the weight is updated. This computational load is almost prohibitive for servers of moderate capabilities. Hence, devising an alternative strategy to mitigate this computational expense is crucial.

4.5.2 Fourier Optics

Next, we aim to leverage Fourier optics (FO) as a means to alleviate the computational burden of the simulation. Prior to discussing the intricacies of the simulation algorithm, it is essential to initially outline the basic principles of FO. This optical theory serves as an approximate representation of the Huygens-Fresnel principle. The following key assumptions underline the principles of FO:

- **A1: Paraxial approximation**. In FO, the assumption $\gamma \approx \sin\gamma$ is made, where γ represents the angle between the optical ray and the Z-axis, as shown in Fig. 4.7(a). The condition for this approximation is that γ should remain small, specifically, $\gamma < 1$ radian. The implication is that the optical beams largely propagate in a manner that is nearly parallel to the Z-axis.
- **A2: Scalar approximation**. FO uses an approximation to transform spherical waves into plane waves in the far field, also known as the Fraunhofer region. This approximation is considered valid when the distance separating the optical source from the plane exceeds ten times the wavelength of the light involved.

FO conceptualizes an optical field as the cumulative effect arising from a multitude

of plane waves, as illustrated in Fig.4.7(b). Optical fields delineate the light intensity's spatial distribution and are sometimes denominated as the spatial spectrum. Within this framework, FO proposes that an optical field can be disaggregated into a sequence of plane waves emanating from distinct angles, referred to as the *angular spectrum*. Transitioning between the spatial and angular spectra of an optical signal is facilitated by FFT and its IFFT [46]. Let $\mathbf{X}(x, y)$ encapsulate the optical field (that is, spatial spectrum) and $\mathbf{A}(f_x, f_y)$ epitomize the angular spectrum, their interconversion adheres to the subsequent equations:

$$\begin{cases} \mathbf{A}(f_x, f_y) = \iint_{-\infty}^{+\infty} \mathbf{X}(x, y) e^{-\mathbf{J}2\pi(f_x \cdot x + f_y \cdot y)} dx dy \\ \mathbf{X}(x, y) = \iint_{-\infty}^{+\infty} \mathbf{A}(f_x, f_y) e^{\mathbf{J}2\pi(f_x \cdot x + f_y \cdot y)} df_x df_y \end{cases} \tag{4.16}$$

where $f_x = \cos\alpha/\lambda$ and $f_y = \cos\beta/\lambda$ serve as angular parameters. As illustrated in Fig. 4.7, α and β are the angles formed between a plane wave's direction and the X-axis and Y-axis respectively. These equations can be articulated in the context of FFT (i.e., $\mathscr{F}(\cdot)$) and IFFT (i.e., $\mathscr{L}(\cdot)$) thusly:

$$\begin{cases} \mathbf{A}(f_x, f_y) = \mathscr{F}\left(\mathbf{X}(x, y)\right) & \text{Spatial} \rightarrow \text{Angular Spectrum} \\ \mathbf{X}(x, y) = \mathscr{L}\left(\mathbf{A}(f_x, f_y)\right) & \text{Angular} \rightarrow \text{Spatial Spectrum} \end{cases}$$

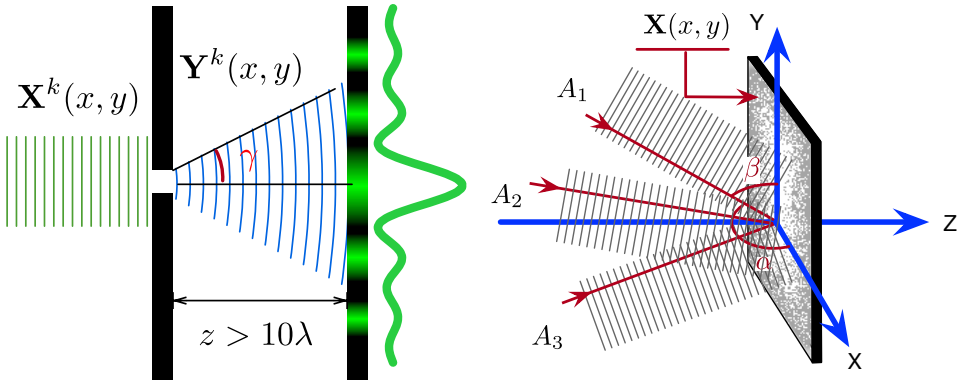
The crux of FO resides in the aforementioned equations.

(1) Simulating Neuron Interaction. FO sheds light on the diffraction dynamics when an optical beam navigates through a neuron. Consider an incoming optical field $\mathbf{X}^k(x, y)$ arriving at the k^{th} layer and passing through a neuron fashioned as a hole, as illustrated in Fig. 4.7(a). The neuron, symbolized by $R(x, y)$, is characterized by two intersecting rectangular functions [8]:

$$R(x, y) = \text{Rect}(x/a)\text{Rect}(y/b) \tag{4.17}$$

with a and b defining the width and height of the rectangle. As a result, the consequent optical field, marked as $\mathbf{Y}^k$, is conceived by the fusion of these functions:

$$\mathbf{Y}^k(x, y) = \mathbf{X}^k(x, y) R(x, y) = \mathbf{X}^k(x, y)\text{Rect}(x/a)\text{Rect}(y/b) \tag{4.18}$$



(a) Diffration (b) Angular decomposition

Figure 4.7: Fourier Optics. (a) The diffraction can be explained by a resulting convolution between the incoming light signal and the square signal in the angular spectrum. The result becomes a sinc signal. (b) We can take FFT on the optical field on a plane to generate a spectrum in which the light is decomposed into a group of plane waves from different angles.

To discern the angular spectrum of the resultant signal, the FFT is applied to $\mathbf{Y}^k(x,y)$:

$$\begin{aligned}\mathcal{A}(\mathbf{Y}^k(x,y)) &= \mathscr{F}(\mathbf{Y}^k(x,y)) = \mathscr{F}(\mathbf{X}^k(x,y)R(x,y)) \\ &= \mathscr{F}(\mathbf{X}^k(x,y)) * \mathscr{F}(R(x,y)) \\ &= \delta(f_x,f_y) * (ab\,\mathrm{sinc}(af_x)\,\mathrm{sinc}(bf_y)) \\ &= ab\,\mathrm{sinc}(af_x)\,\mathrm{sinc}(bf_y)\end{aligned} \tag{4.19}$$

where $\mathcal{A}$ denotes the angular spectrum of the optical field and $*$ is the convolutional operation. The process is underpinned by the Fourier transform principle: in the spatial domain, the merging of two signals corresponds to their convolution in the angular domain. When subjected to the FFT, $\mathbf{X}^k(x,y)$ manifests as a Dirac function, whereas the outcome of $R(x,y)$ is a pair of distinct Sinc functions. Their convolution thus yields two Sinc functions.

The convolution result suggests that the neuron modeled as a hole acts as a secondary emitter, redistributing the optical signal across various angles with different power

levels. This is visually demonstrated in Fig. 4.7(a). The resultant optical field on the subsequent plane is essentially a manifestation of these projected Sinc functions. This accounts for the alternating bands of light and darkness observed, with their intensity diminishing as one moves away from the center. Additionally, the width of the primary lobe of the Sinc function is inversely correlated with the dimensions of the opening, mathematically expressed as $\Delta\theta_x \approx 2\lambda/a$ and $\Delta\theta_y \approx 2\lambda/b$. To meet the assumption A1, $a = b = \lambda$.

(2) Simulating Inter-layer Light Propagation. Next, we apply FO to simulate the propagation of light between two parallel planes separated by a distance z. The optical fields at these planes are denoted as $\mathbf{Y}^k(x, y)$ and $\mathbf{X}^{k+1}(x, y)$, as illustrated in Fig. 4.8. Their relationship can be mathematically represented as [45] :

$$\begin{aligned}\mathbf{X}^{k+1}(x,y) &= \frac{e^{\mathbf{J}kz}}{\mathbf{J}\lambda z}\iint_{-\infty}^{+\infty}\mathbf{Y}^k(x_0,y_0)e^{\mathbf{J}\frac{\pi}{\lambda z}\left((x-x_0)^2+(y-y_0)^2\right)}dx_0dy_0 \\ &= \iint_{-\infty}^{+\infty}\mathbf{Y}^k(x_0,y_0)h^k(x-x_0,y-y_0)dx_0dy_0\end{aligned} \tag{4.20}$$

where $h^k(x, y)$ is termed the channel parameter:

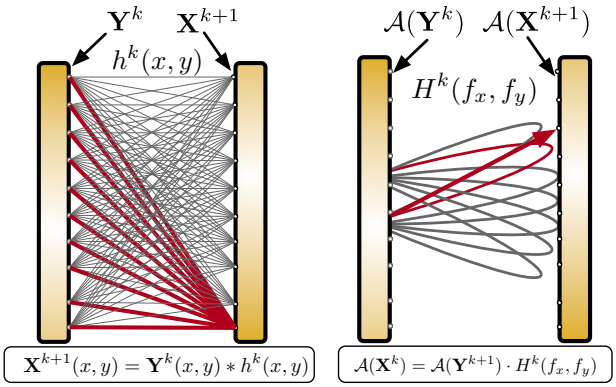
$$h^k(x,y) = \frac{e^{\mathbf{J}kz}}{\mathbf{J}\lambda z}e^{\mathbf{J}\frac{\pi}{\lambda z}(x^2+y^2)} \tag{4.21}$$

The channel function $h^k(x, y)$ approximates the attenuation in amplitude and rotation in phase that occur along the propagation path (see Eqn. 4.5). The field $\mathbf{X}^{k+1}(x, y)$ can be viewed as the convolution of $\mathbf{Y}^k(x, y)$ and $h^k(x, y)$:

$$\mathbf{X}^{k+1}(x,y) = \mathbf{Y}^k(x,y) * h^k(x,y) \tag{4.22}$$

Fig. 4.8(a) visually shows this procedure where the red lines are the channels for one step of the convolution. In the figure, $\mathbf{Y}^k(x, y)$ remains fixed, but the channels are shifted from top to down (and left to right). Each shift is to compute the optical signal received at a point on the second plane.

The property of the Fourier transform suggests the convolution of two optical signals in the spatial domain is equal to their dot product in the angular domain. Thus, the



(a) Spatial domain (b) Angular domain

Figure 4.8: Inter-layer propagation. (a) Fourier optics considers the light propagation between two parallel planes as a result of convolution between the optical field and the channel function. (b) The light propagation can also be viewed as a result of their dot product in the angular spectrum

above computations can be reduced as follows:

$$\mathcal{A}(\mathbf{X}^{k+1}) = \mathcal{A}(\mathbf{Y}^k) \cdot \mathcal{A}(h^k(x,y)) = \mathscr{F}(\mathbf{Y}^k) \cdot \mathbf{H^k}(f_x, f_y) \tag{4.23}$$

where

$$\mathbf{H}^k(f_x, f_y) = \mathscr{F}(h^k(x,y)) = e^{\mathbf{J}kz} e^{-\mathbf{J}\pi\lambda z(f_x^2+f_y^2)} \tag{4.24}$$

Fig. 4.8(b) visually explains why the convolution becomes a dot product in the angular spectrum. The received optical signal at a particular point on the second plane corresponds to the angular signal modulated by channel attenuation towards that point. FO provides an efficient computational framework for simulating light propagation, thereby paving the way for new and expedited training algorithms in the context of signal processing.

4.5.3 Simulation Algorithm

We outline the simulation algorithm used to compute $|\mathbf{X}^K|$ – the intensity distribution of the final layer, essentially representing the image detected by the camera.

- **Step 1: Initialization**. We initiate with setting $\mathbf{X}^0 = \text{ones}(\sqrt{N}, \sqrt{N})$, wherein $x_i^0 = 1$. Here, each light beam reaching the input layer is assumed to have a unit amplitude with zero phase.
- **Step 2: Beams out of Neurons**. For any incident optical beams, $\mathbf{X}^k$, that converge on the k^{th} layer, the resultant beams, $\mathbf{Y}^k$, after traversing through this layer can be deduced as:

$$\mathbf{Y}^k = \mathbf{X}^k \odot \mathbf{W}^k \tag{4.25}$$

  where $\mathbf{W}^k$ represents the binary weight matrix, while $\odot$ symbolizes Hadamard product.
- **Step 3: Beams into Neurons**. With the emergent beams $\mathbf{Y}^k$ exiting the k^{th} layer, the next layer's beams, $\mathbf{X}^{k+1}$, are computed as:

$$\mathbf{X}^{k+1} = \mathscr{L}(\mathcal{A}(\mathbf{X}^{k+1})) = \mathscr{L}(\mathscr{F}(\mathbf{Y}^k) \cdot \mathbf{H}^k) \tag{4.26}$$

  The optical field $\mathbf{Y}^k$ undergoes transformation into its angular representation $\mathcal{A}(\mathbf{Y}^k)$ via FFT. Subsequently, $\mathcal{A}(\mathbf{Y}^k)$ gets multiplied by the channel's angular form to yield the angular representation for $\mathbf{X}^k$. Eventually, we revert to the spatial representation of $\mathbf{X}^k$ using the IFFT.
- **Step 4: Intensity distribution**: Steps 2 and 3 are iteratively executed for every layer until the beams arrive at the terminal layer. The resultant output intensity distribution is then determined as $|\mathbf{X}^K|$.

Each time an input is presented, we employ the aforementioned simulation algorithm to ascertain the output and juxtapose it with the labels (i.e., the anticipated distribution) to compute the loss, which is further used for the error back-propagation.

Leveraging FO, this entire process harnesses the power of FFT and IFFT, thus reducing the computational complexity from $\mathcal{O}(KN^2)$ to $\mathcal{O}(KN\log(N))$.

4.6 Implementation

4.6.1 Fabrication Techniques

The three types of layers are made using different techniques: • **Hidden Layers**. The essence of BONN lies in its hidden layers. Each hidden layer contains a matrix of $1,000 \times 1,000$, amounting to 1 million neurons. These neurons are circular apertures, each having a diameter of 800 nm. Consequently, the overall size of a hidden layer is approximately 0.8×0.8 mm^2. Additional space is reserved as margins to ensure secure mounting. A microscopic view of a hidden layer is presented in Fig. 4.2. The Photolithography Equipment (PE) technique is employed to fabricate a layer on a foundation of silicon dioxide. We carefully coated each neuron layer with a light-absorbing black material-Cr layer, which is 120 nm thick, to block light propagation if the weight is 0 and kept areas corresponding to weight 1 transparent. Cr layer will block 93% [4] incoming lights on average according to our experiments. In practical applications, 18 distinct layers are meticulously etched for six BONN configurations, with each model housing three of these hidden layers, all on a solitary substrate to further economize on production costs. To save cost, up to 18,241 $0.8 \times 0.8\text{mm}^2$ hidden layers can be etched on a single 6-inch wafer, which has a production cost of 2,358 USD. This approach brings down the expense to a mere 0.13 USD per layer.

• **Input Layer**. The resolution of the input layer in our system is set as the same as neurons, ensuring that each neuron receives one pixel of the input image. This input layer is pre-manufactured with target images, which are then replaced as needed for different tasks manually. While this approach offers a temporary measure for evaluating the viability of BONN, future advancements aim to integrate SLM for

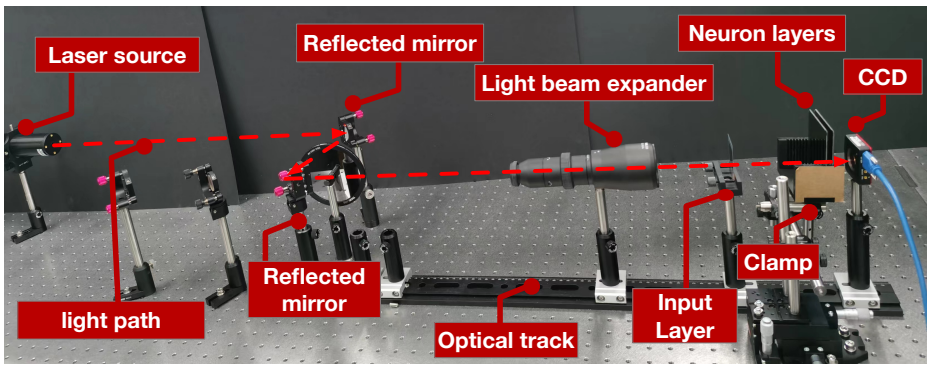


Figure 4.9: Illustration of the experimental setup

compact and dynamic input handling.

• **Output layer**. The output layer utilizes a CMOS camera with a resolution of 1440×1080 pixels [3]. Each pixel occupies a square area with a side length of $3.45\mu m$. The camera's pixel array is designed to fully cover the output. As mentioned earlier, the $0.8 \times 0.8\text{mm}^2$ layer is divided into broader sections, each representing a specific classification outcome. Thus, even though the size of the camera's pixel exceeds that of a single neuron, the camera remains proficient in accurately capturing the output results.

Table 4.1: Datasets Settings and Performance.

Dataset	Training #	Testing #	Class #	Accuracy			
				Simulation	Prototype	BENN	SOTA
MNIST [63]	60,000	10,000	10	0.87	0.74	0.89	0.997 [25]
Fashion [120]	60,000	10,000	10	0.776	0.66	0.782	0.982 [107]
English [10]	5,463	1,065	26	0.844	0.75	0.85	0.901 [2]
Greek [75]	3.626	487	10	0.715	0.59	0.72	0.833 [109]
Oracle [53]	10,387	1,708	10	0.716	0.57	0.712	0.787[53]
Posture [61]	3,839	962	4	0.809	0.72	0.813	0.879 [125]

4.6.2 Datasets

Given the inherent nature of BONN, where weights are modulated by passive devices incapable of online learning, we have to adopt a two-step approach. Initially, BONN is simulated in software, termed the "simulation model". Once optimized, the weights are then deployed in real-world scenarios, referred to as the "prototype model". BONN is trained using the PyTorch framework [87]. Each model is trained at a standard server powered by an AMD 5900x (4.9GHz) processor, complemented by 64GB RAM and dual NVIDIA 3080Ti GPUs. Stochastic gradient descent is used to optimize the training process with a learning rate of $1 \times e^{-2}$ and a momentum of 0.9 across the whole experiment. The batch size is set as 128. The training epoch is 400. The resolution of images is 400×400 pixels. The prototypes are tested across six public datasets, including a total of 67,537 data samples, of which 43,315 samples are used for training, and the remaining 24,222 samples are for testing. The detailed information about the six datasets is listed in Table 4.1. They are the MNIST, Fashion, handwritten English characters, handwritten Greek characters, the posture of humans, and the Oracle.

4.6.3 Experimental Setup

The prototypes are evaluated using an optical table, as shown in Fig. 4.9. HNL05LB HeNe laser obtained from Thorlabs Inc., featuring a power output of 5mW and priced at $1837.5, is utilized to generate 632.8nm coherent lights. Importantly, the laser source did not have specific mandated specifications, providing us with the flexibility to employ the same laser for all tasks. Two reflector mirrors align the optical rays emanating from this source, ensuring that only parallel light rays continue forward, while others are deflected at different angles. A beam expander then broadens this narrow ray [1]. All physical layers, including the input layer and the hidden layers, are fixed and aligned along a straight line using the clamp. The CMOS camera acts as the

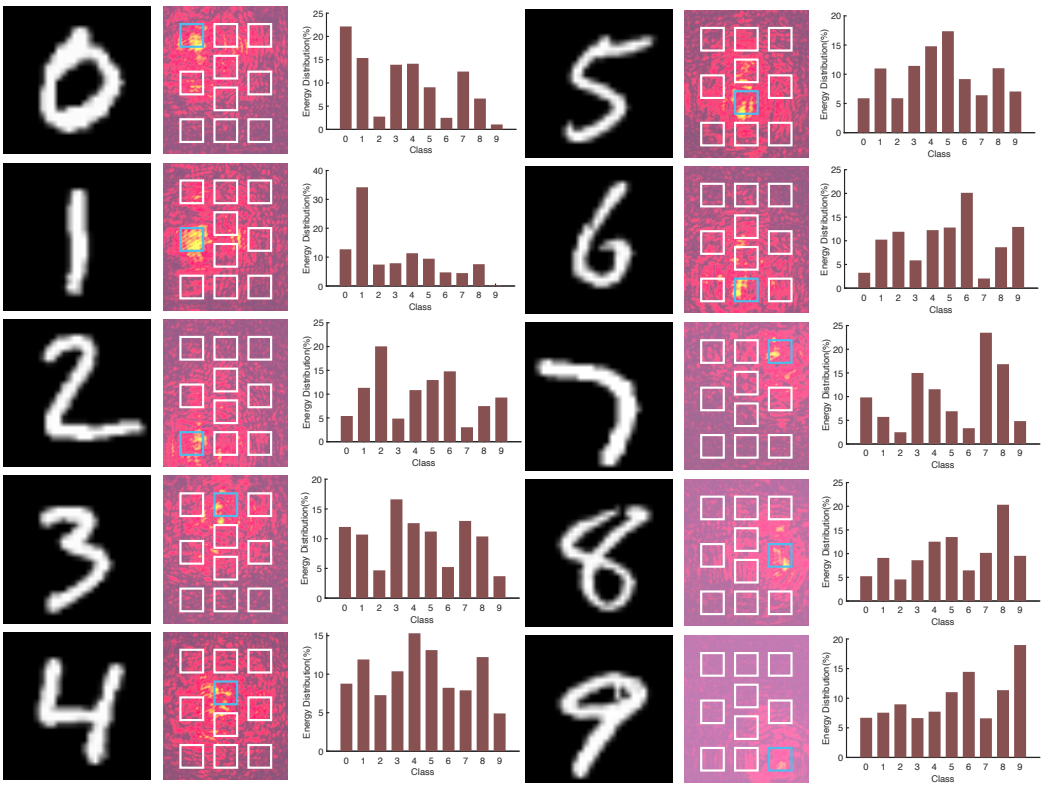


Figure 4.10: Result demonstration of MNIST. The left, middle, and right columns show the input digital, captured output picture and the energy histogram, respectively.

output layer. The spacing between adjacent hidden layers is set at $1.5mm$, denoted as z in Eqn. 4.24. The diameter of the diaphragm is $25mm$. As aforementioned, Fourier optics operates under two assumptions. First, the space between hidden layers is fixed at $1.5mm$, which is far beyond 10λ, so it meets the scalar approximation. Second, the paraxial angle is $\arctan(0.88\text{mm}/1.5\text{mm}) = 0.5\text{rad} < 1$ rad, so it also meets the paraxial approximation. Thus, the experimental settings meet the two basic assumptions.

4.7 Benchmark

We start with benchmark experiments with the prototypes to provide insights into the working of BONN.

4.7.1 Feasibility

First, the output from the prototyped BONN, captured by the CMOS camera, is demonstrated in Fig. 4.10. The captured images exhibit a light red hue because 632.8nm light is adopted as the source. They are finally converted to gray-value images. The entire image (i.e., output layer) is divided into 10 disjoint regions, with each region signifying a specific classification result. The guard bands between two adjacent regions are deliberately reserved to avoid cross-region ambiguity. Pixel gray values are interpreted as energy levels, and the cumulative energy within each region is tallied. The energy histograms across the 10 regions are shown on the right. The classification is determined based on the region with the peak energy. Surely, we also meet the failures in which all bars are lower than 10%, i.e., each bar has quite even energy. In this case, the input is deemed indistinguishable. These results successfully showcase the feasibility of using BONN for classification tasks. Given that the classification is based on region energy which is associated with light intensity, BONN is vulnerable to noise arising from non-coherent lights which may cause incorrect power level readings on the output layer. BONN is tested in a dark environment to reduce the noise and help mitigate the low intensity problem.

4.7.2 Overall Accuracy

Second, the accuracy of the prototyped BONN is evaluated across the six datasets: MNIST, Fashion, English, Greek, Oracle, and Posture. These datasets vary in terms of complexity, size, and number of classifications. The accuracy results are listed in

Table 4.2: Comparision with SOTA ONN. Tested on the MNIST dataset.

System	Tech.	Cost/Layer	Neurons #	Wavelength	Acc.
D2NN [68]	3D-Print	100 USD	0.4 M	0.75 mm	0.845
SONN [130]	SLM	13K USD	225K	698 nm	0.923
FONN [97]	SLM	13K USD	1 M	780 nm	0.33
BONN	PE	0.13 USD	1 M	632.8 nm	0.74

Table. 4.1. Overall, BONN achieves mean accuracy of 67%. BONN performs best in recognizing the handwritten letters and MNIST dataset (i.e., digitals) with the accuracies of 75% and 74%, respectively. It performs worst in recognizing the Oracle characters, where the accuracy is as low as 57%. The Oracle characters present a unique challenge because they are too primitive and figurative. Some similar Oracle characters might represent different meanings when being looked at from different angles, and this is one of the reasons why the Oracle characters are gradually replaced by modern non-figurative characters. Our results show an about 20% accuracy rise in identifying modern characters than Oracle ones. In addition, Oracle characters were obtained after thousands of years, so the majority of them are of poor quality than the two other datasets, which also seriously affects the recognition accuracy. BONN achieves a relatively higher accuracy of 72% in recognizing the posture and fashion datasets, likely due to their restriction to just four classes.

4.7.3 Comparison to ONN

Third, we compare BONN with the state-of-art ONN. Results for other ONN systems are derived from the corresponding report due to the lack of equipment. D2NN [68], SONN [130] and FONN [97] are selected as the baselines. Their configurations and prototype accuracy are listed in Table. 4.2. Our observations are as follows: **(1) Cost**. BONN is ranked first without a doubt. Its per-layer cost is 0.13% of D2NN, a phase-adaptive all-optical ONN. The cost-effectiveness of BONN is due to two reasons. First,

PE is widely adopted as a rather mature technique for chip manufacturing compared with the spatial light modulator(SLM), so scale production makes it cost-effective. Second, the binarized neurons only require etching the substrate once, which greatly streamlines the manufacturing. The current version of BONN needs refabrication when confronted with new datasets or tasks, which is the key limitation for real-world deployment. In the future, we may consider training a large-scale backbone BONN, drawing inspiration from the success of expansive AI models like ChatGPT. This backbone BONN can be fine-tuned for specific tasks by incorporating additional layers, offering a scalable solution with minimal cost implications. **(2) Accuracy**. Certainly, excellent cost control is at the price of accuracy. D2NN and SONN achieve mean accuracies of 84.5% and 92.3%, which are 15% higher on average than that of BONN. Both previous works adopt the weight-adjustable approach in which the phase shift varies between $0° - 360°$. Unfortunately, they only work at THz frequency instead of at the visible spectrum, leading to a larger physical size, fewer neurons, and unavailability for image processing. **(3) Availablity**. Controlling visible light is much more difficult than THz because of its shorter wavelength. FONN is the recent effort to use an active metasurface (i.e., spatial light modulator) to adjust the phase for visible light. However, their real experiment results are unsatisfactory (i.e., 33%), although the author claims a 90% above accuracy in the simulation. In summary, BONN is the first to achieve relatively good accuracy in the visible spectrum at an extremely low cost. **(4) Fabrication:** PE boards are less flexible with fixed parameters, while SLMs can dynamically update network parameters. However, PE boards in BONN have higher transmittance, achieving 98-99% [7] light transmission, versus SLM's 93% [9]. Moreover, PE boards offer a nanometer-level resolution, superior to SLM's micrometer-level.

Table 4.3: Comparision with SOTA ENN.

System	Power	Accuracy	Time (us)
FPGA-BNN [67]	8.9 mJ	0.98	340
MCU-BNN [42]	40 mJ	0.99	1000
GPU-BNN	2711 J	0.92	8254
BONN	3.7 μJ	0.74	0.008

4.7.4 Compared to ENN

Fourth, BONN is compared with ENN. A GPU-based BNN with the same architecture is set up as the baseline. It is trained and tested on a PC with an Nvidia 3080 Ti GPU. BONN is also compared with two SOTA low-power TinyML models, namely, an MCU-based BNN model [42] and an FPGA-based BNN system [67]. All systems are trained and tested on the MNIST dataset. The results are listed in Table. 4.3. The power and interference time is the average consumed by processing a single input image. **(1) Power**. A GPU power meter is used to measure the consumption of ENNs. The most low-power ENN is the MCU-BNN, which consumes about 40mJ to process a single image. In BONN, the only power-consuming components are the light source (i.e., 5mW), the CMOS camera (i.e., 1.5 μW), and the host system that analyzes the image from CCD (i.e., 3.6W). Regarding $40\mu s$ exposure time, BONN totally consumes (5mW + 1.5μW) $\times$ 40μs + 3.6 $*$ 0.001/1024s = 3.7μJ, which can be further reduced by using LED array instead of the camera. By contrast, even regarding the most power-saving ENN (i.e., 8.9 mJ FPGA-BNN), BONN consumes $2,405\times$ less power. As mentioned earlier, ONN is much more power efficient than any ENN. **(2) Speed**. BONN runs at the speed of light, so the time is mainly determined by how long it travels from the light source to the output sensor. The total length of the optical path is about 2.4m. By contrast, the logical ENNs usually need additional operations such as register shifting, cache paging, onboard communication, etc. Thus, BONN saves time by 99.7% compared with the fastest ENN (i.e., FPGA-BNN). **(3) Accuracy**. In terms of the MNIST dataset, the mean accuracy of the ENN models

is around 95%, which is 20% higher than that of BONN. As discussed later, the main error source comes from the manual experimental setup.

Modern low-power platforms like MCUs are highly energy-efficient and suitable for many applications. However, their capabilities are limited in scenarios requiring intensive real-time data processing, such as high-dimensional beamforming or large-scale matrix computations. ONNs offer unique advantages in such cases, including massive parallelism, lower power consumption per operation at scale, and significantly reduced latency due to their photonic nature. ONNs are not intended to universally replace MCUs but to address the specific demands where traditional platforms struggle to meet performance and efficiency requirements.

4.7.5 Comparison to Simulation

Fifth, the performance is compared between the simulation and the prototype. **(1) Distribution**. Fig. 4.11 shows the output layers and the energy distributions acquired via the simulation and the prototype when recognizing the handwritten digital one. Visually, the energy concentrates more on the target region in the simulated output layer than that on the prototyped layer. Specifically, 34% and 28% of energy fall into the target region in the simulated and prototyped results, respectively. Before testing, we must manually align the layers along a straight line seamlessly on the optical platform without a reliable calibration method. Such manual operation is challenging. Modern nano-level packaging technology offers potential solutions to this challenge [66]. **(2) Accuracy**. The recognition accuracy of the simulations and the prototypes across the six datasets are compared. The results are listed in Table 4.1. The mean accuracy results taken by the simulation and prototypes are 81% and 67%, respectively. The simulation achieves 14% higher accuracy than the prototype. The energy loss mentioned above mainly accounts for the accuracy decline. Additionally, each class is tested with around 1000 samples, while the prototypes are tested with

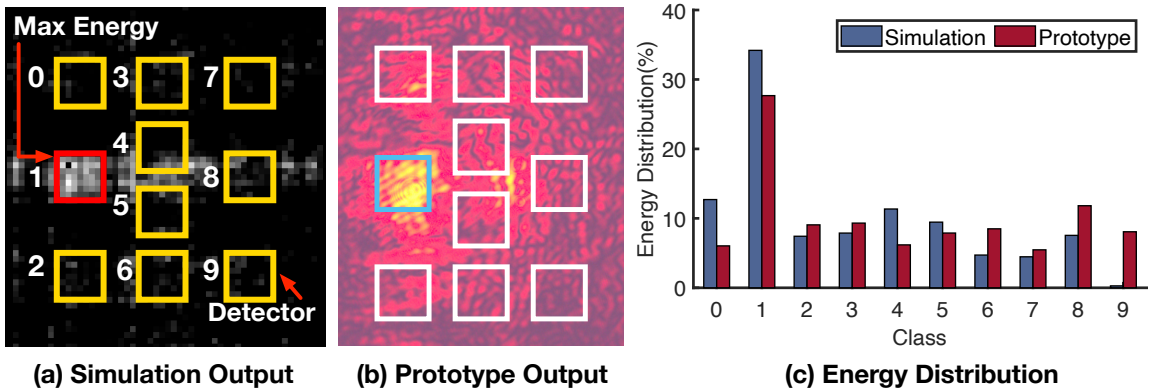


Figure 4.11: Simulation Based Training. (a) and (b) are the output image of the simulation and prototype ; (c) shows the are the energy distribution of the output layer.

about 10 samples. Thus, a part of the decline also arises from the limited number of prototype samples. One might be concerned that why do not test more? We made 10-15 input layers for each class from a dataset, so each class is tested by only ten samples. Regarding 70 classes from the six datasets, a total of 1000 input layers are tested via manual operations, which is a heavy workload for us already.

To enhance BONN's accuracy and bridge the gap between simulations and prototypes, we outline three significant avenues for future work: Firstly, we can refine our training process by incorporating advanced tuning techniques, such as an adjustable learning rate and meticulous initialization of trainable parameters. These are successful practices in other tasks [21] and are expected to significantly improve the accuracy of BONN. Secondly, we can further address misalignment errors by integrating BONN into an on-chip camera system, as demonstrated in prior work [73]. This integration not only enhances BONN's accuracy but also promotes synergy with existing mobile systems, facilitating portability without altering the model or its passive mode while maintaining energy efficiency and cost-effectiveness. Thirdly, we can also design a hybrid model combining BONN and ENN to enhance performance and address scalability concerns.

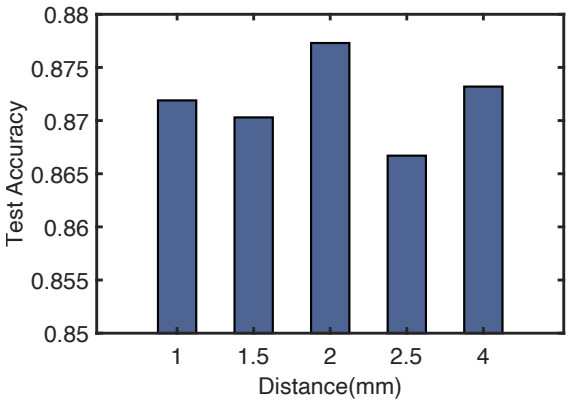


Figure 4.12: Impact of layer space

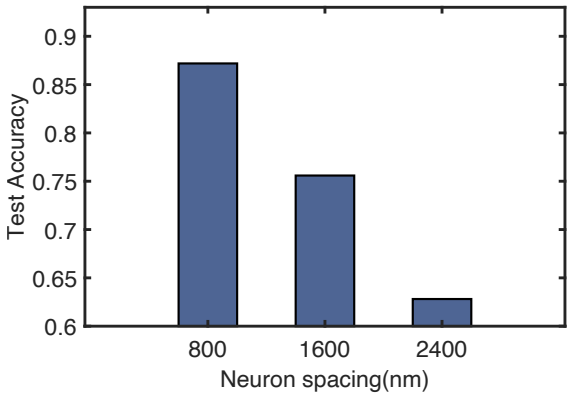


Figure 4.13: Impact of neuron space

4.7.6 Comparision to Binary ENN

Sixth, we compare BONN with Binary ENN(BENN) which is binarized by STE [33]. BENN, featuring two fully connected layers with 400 neurons, is trained across the experiment with a learning rate of 1×10^{-4}, momentum of 0.9, batch size of 128, and for 400 epochs. The results are listed in Table 4.1. The mean accuracy over six datasets is 82%, which is 1% higher than that of the simulated BONN. BENN performs well on the MNIST, English and Posture datasets like the simulated BONN model with the accuracy 89%, 85% and 81.3%. And it performs worst on the Oracle dataset and the accuracy is 71.2%. The primitive and figurative Oracle also presents a huge challenge for BENN. The observations prove that the light propagation model of BONN is well-matched with the BENN.

4.8 Results

Finally, we conduct a large-scale evaluation of the impact analysis and the scalability via the simulation regarding the fact that the simulation performs comparably as prototypes.

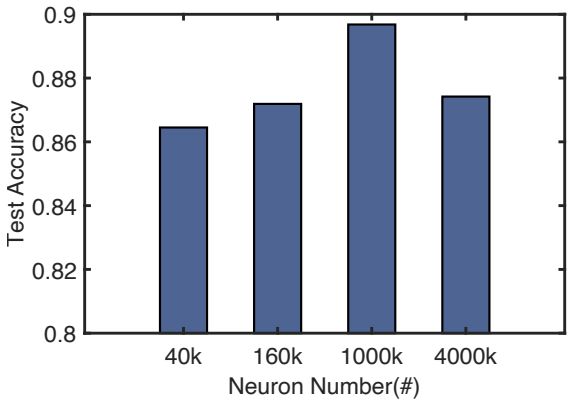


Figure 4.14: Impact of neuron number

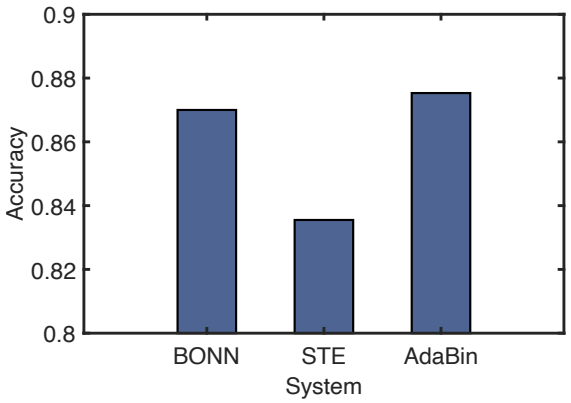


Figure 4.15: Impact of binary functions

4.8.1 Impact Analysis

We test the four types of impacts on the performance of BONN regarding the MINIST dataset as follows:

(1) Impact of the space between layers. As introduced earlier, the biases of an ONN are determined by the space between two adjacent layers. We tested the 1-4mm settings with a step of 0.5mm. For each setting, we need to train a new model. The results are shown in Fig. 4.12. As a result, the accuracy varies within 0.5%. Thus, space is not a key factor for BONN because the impact on the neurons which share the space might cancel each other out

(2) Impact of space between neurons. By default, two adjacent neurons are spaced at 800 nm, which is the distance between the two neurons' centers. We also test the other two settings where the space is set to 1600nm and 2400nm. The results are shown in Fig. 4.13. It can be seen that the accuracy rapidly decreases to 75% and 64.5%. The double-slit interference theory suggests that a wider space makes the energy of the interfered results more distracted at a wider section, which is against the paraxial assumption that the majority of energy should be concentrated on the ± 1 rad. Thus, we desire a smaller space.

(3) Impact of the neuron number. Next, we test the impact of the number of

neurons on a single layer. We tested four types of networks. Each of their hidden layers contain 200×200, 400×400, 1000×1000, and 2000×20000 neurons, respectively. The results are shown in Fig. 4.14. Clearly, more neurons show higher learning ability, so the accuracy usually increases as the number of neurons. However, when the number reaches 4000k, the network is so sensitive that being overfitted where noise is learned as well, leading to the accuracy decrease. This is a common phenomenon in deep neural networks.

(4) Impact of different binary functions. Then, we compare the performance across different binary functions. The comparative analysis involves benchmarking against STE[33] and AdaBin[110], illustrated in Fig. 4.15. The accuracy of STE, BONN, and AdaBin is 0.8355, 0.87, and 0.8753, respectively. AdaBin takes the lead but only outperforms BONN by a tiny bit in accuracy. In practical prototype scenarios, this difference may even be subtle. Thus, Gumbel-sigmoid in BONN is sufficient to binarize the weights in the light propagation paradigm.

(5) Impact of the number of hidden layers. Fig. 4.16 shows the accuracy as a function of the number of hidden layers including 1 million neurons. Fig. 4.17 plots the mean light intensity of each layer compared with the "0" layer, representing the input layer. We achieve maximal accuracy when adopting two hidden layers. More layers result in the accuracy decrease even if the total number of neurons is increased. This result is consistent with our previous analysis; that is, the depth of ONN does not help improve performance.

The reason, as depicted in Fig. 4.17, is the significant attenuation of light intensity in deeper layers of the network, with the loss exceeding 50% in each successive layer. Thus, a BONN with two or three layers is highly recommended for optimal performance. However, a deeper BONN architecture is feasible where middle results captured by CCD are relayed to subsequent layers with amplification to compensate for the light attenuation.Additionally, an alternative in the future is to leverage a light splitter to divide the light into two distinct beams. One beam undergoes direct

diffraction, while the other emulates a ResNet-inspired skip connection, serving as immediate input for the subsequent layer. This approach may effectively circumvent layer attenuation.

4.8.2 Real-Life Application: Face Recognition

Finally, we explore the potential of BONN in a complicated real-life application – face recognition. We train a BONN on two publicly available face datasets, CK [57] and CK+ [72]. The dataset contains 8,815 samples and 100 classes (i.e., 100 subjects' faces), wherein 7,006 samples are used for training, and the remaining 1,809 samples are used for testing. To accomplish this task, we segment the output layer into 100 regions, with each region spanning an area of 20×20 pixels. The energy distribution is acquired in different regions as shown in Fig. 4.18. Visually, the energy concentrates more on the target region than on other regions. As a consequence, BONN achieves 83% recognition accuracy, aligning closely with the performance metrics observed in our previous six datasets. This result demonstrates the capability of BONN for complicated classification tasks. Despite an anticipated drop in accuracy from simulations to real-world applications, the promising results suggest our system's potential for effective deployment. Its rapid processing and low energy consumption make it ideal for high-volume face recognition like crowd monitoring. Additionally, our system can enhance detection efficiency by cooperating with the traditional ENN in a cascade mode.

4.8.3 Potential Application Scenarios

Although BONNs have lower accuracy than ENNs on some datasets, their energy efficiency and speed are advantageous for high-volume tasks or we can use BONNs as part of a cascade. We outline them as below: 1) Preliminary Screening: Effective for initial anomaly detection in large datasets, useful in environmental monitoring

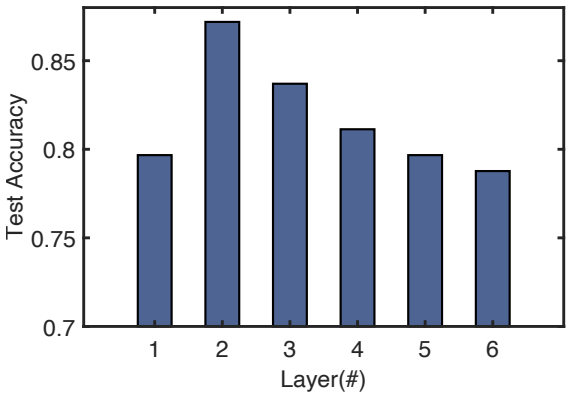


Figure 4.16: Impact of layer number

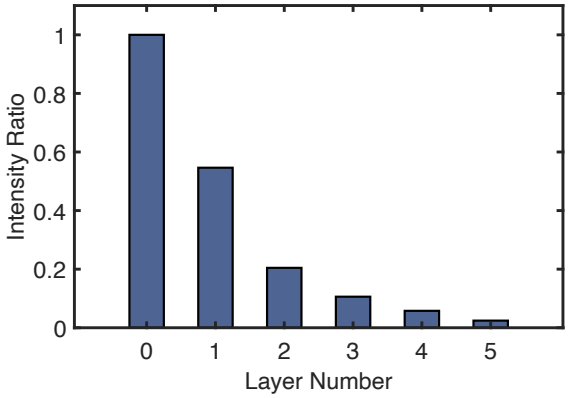


Figure 4.17: Layer intensity ratio

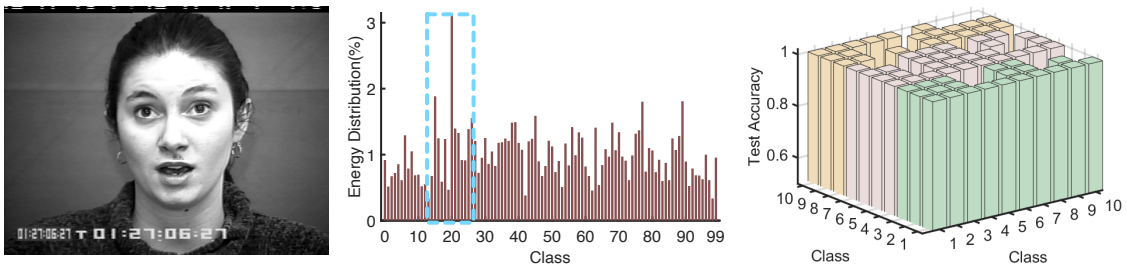


Figure 4.18: Face recognition task. There are the input, the energy distribution, and the classification results over the dataset.

or security. 2) Real-time Applications: Offers low latency for real-time detection or tracking while economizing on energy. 3) Energy-limited Devices: Enhances longevity of devices like sensor networks or wearables. 4) Remote Area Applications: Facilitates rudimentary processing where energy or connectivity is limited. 5) Pre-processing and Data Compression: Mitigating data transmission demands by extracting typical information prior to cloud server transmission. 6) Assistive Decision-making: Offering faster, though slightly less accurate, decision support to speed up decision-making paradigms. For precise needs in critical applications, a hybrid approach combining ENN with ONN for high accuracy tasks is recommended which harmoniously balances accuracy with energy use.

4.9 Conclusion

This work introduces BONN, an evolution of ONN that bridges the gap between software-driven methodologies and practical implementations. By leveraging the binarization method, we've not only simplified the fabrication of ONN systems but also championed the use of diffraction effects to facilitate computations at light speed. While there are slight compromises in accuracy, the notable gains in computational speed and power efficiency are remarkable.

4.10 Related Work

BONN intersects with prior research in three areas: optical neural network, binarized neural network, and metasurface.

Optical Neural Network: Recently, the topic of all-optical neural networks has garnered considerable attention from the AI community given its sustainability and scalability [68, 24, 131, 121, 80, 43, 48, 77, 118, 99, 130, 26, 73]. In 2018, [68] first presents the idea of the ONN and implements a prototype by using phase controllable diffraction elements. However, due to its fabrication limitation, the physical size of its emulated neuron is around a millimeter, which is too large to be scalable and can only work on the invisible infrared band. The following research has been done to shrink neuron size and improve its scalability by using hybrid optical-electronic design [24], photoelectric multiplication [48], laser printing [43] and metasurface [73]. These systems rely on extremely complex fabrication techniques and hence, are too expensive to be practical. State-of-the-art work such as MDNN [73] reports an ONN working in the visible range by using an advanced metasurface technique. However, it requires an extremely expensive EBL fabrication process, and spreading its application is difficult. Unlike past works, BONN first introduces binarization into the all-optical neuron networks, which substantially simplifies the ONN fabrication and

makes it more practical. Some ONNs[103, 126] operate at an invisible light range with active devices such as Mach-Zehnder Interferometers which consume much more power than passive devices.

Binarized Neural Network: Binarization is a promising technique to save the neural network computation memory, time, and energy by using binary network parameters instead of floating-point parameters, which is widely applied to resource-limited devices [54]. The key to binarization is to find an adequate sign function to binarize the network weights. Past works propose numerous binarization sign functions, including a fixed sign function [32], adaptive scaling sign function [92], loss-aware binarization [50], distributed activation and approximate sign [70]. Some works[103] utilize the straight-through estimator (STE) for gradient approximation, a meticulous training method that may reduce accuracy greatly after binarization. Contrary to previous BNNs, BONN necessitate a balance between open and closed neurons to prevent information loss while maintaining accuracy after binarization. This issue is solved by introducing the Gumbel-Sigmoid function.

Chapter 5

Conclusion and Future Work

5.1 Conclusion

To fully understand the significant advancements discussed later, it's crucial to first grasp the key challenges that have historically impacted RFID systems and optical neural networks. Understanding these obstacles provides a necessary foundation for recognizing the progress made in these fields. Specifically, our exploration into the realms of RFID technology introduces Transfer Beamforming (TBF), a novel strategy designed to overcome the significant limitations of power supply and efficient data transmission within battery-free backscatter devices. Simultaneously, our investigation into beamforming in optical computing leads to the development of Binary Optical Neural Networks (BONNs), which represent a strategic effort to significantly mitigate the costs and energy demands associated with traditional optical neural networks. To be specific, our contributions can be summarized as:

• **Transfer Beamforming via Spatial Relationships:** In this work, we address the significant limitations faced by the widespread deployment of battery-free backscatter devices such as RFID tags, specifically their short reading ranges and high miss reading rates due to inefficient energy harvesting. The conventional solution of beam-

forming encounters a deadlock problem where sufficient power is necessary to activate the backscatter, which in turn is essential to provide the channel parameters needed for effective beamforming. To overcome this challenge, we introduce a novel concept called Transfer Beamforming (TBF). This innovative approach involves using semi-active tags as reference points that are powered by standard readers and can then assist in directing beamforming energy to activate surrounding passive tags more effectively. Our development of TBF demonstrates a practical solution to enhance the performance of RFID systems by leveraging known positions of reference tags to optimize the beamforming process for unknown adjacent tags. By implementing this strategy, our prototype evaluation achieved an impressive 99.9% inventory coverage in a densely populated warehouse environment, with a significant improvement in power transmission by 6.9 dB and a doubling of inventory processing speed compared to existing methods. These advancements not only showcase the effectiveness of TBF in practical applications but also mark a pioneering step in utilizing beamforming technology to enhance the capabilities and efficiency of backscatter communication systems. This approach not only mitigates the initial limitations of RFID technology but also opens up new avenues for improving wireless energy transfer and communication precision in complex environments.

• **Binary Optical Neural Networks with Million-Scale Neurons:** In this study, we tackle some of the inherent limitations of electronic neural networks (ENNs), notably their high energy consumption and limited computational speeds, by leveraging the potential of optical neural networks (ONNs). Despite the advancements in ONNs, their widespread adoption has been hindered by high manufacturing costs. To address this challenge, we introduced a novel model: the Binary Optical Neural Network (BONN), which utilizes binarized weights to significantly simplify and reduce the cost of fabrication. This streamlined version of ONNs, BONN, employs a binarization weight function compatible with backward-error propagation and a simulation-based training method that caters to large-scale networks housing millions of neurons. By

reducing costs to $0.13 per layer which is a drastic reduction by 769 times compared to traditional ONNs and cutting power consumption by 2,405 times relative to ENNs, BONNs present a viable and efficient alternative. Furthermore, BONNs maintain a good recognition accuracy of 74% across diverse datasets. The training of BONNs, which integrates a Fourier Optics (FO) enabled approach, converts complex spatial domain convolutions into simpler dot product operations, thereby reducing computational demands. This innovation makes a difference by integrating binary neural network principles into optical settings at visible wavelengths, paving the way for their potential inclusion in portable devices. Our prototypes confirm that BONNs are not only feasible but also perform comparably in accuracy and reliability, setting the stage for broader application and further exploration in optical computing technologies.

5.2 Future Work

This thesis has focused on advancements in beamforming for wireless communication and optical computation, showcasing a range of innovative solutions that enhance signal accuracy and system efficiency. While considerable progress has been made, the increasing demands of modern technology continue to drive the need for further improvements. Future research will aim to refine current methods and introduce new approaches that could significantly improve beamforming techniques in both wireless and optical domains.

As we continue to advance and expand the capabilities of beamforming in both wireless communication and optical computation, ongoing research is crucial for achieving not only improved performance and reliability but also for addressing the growing energy demands of digital infrastructures. This section will highlight key areas where innovations in beamforming could significantly enhance the efficiency and functionality of wireless and optical systems, contributing to more sustainable technology

solutions worldwide.

5.2.1 Simplified Beamforming System

In TBF, the deployment of 150 semi-active tags is initially seen as a one-time effort. However, the ongoing commitment to reduce the reliance on these semi-active tags opens up transformative possibilities for simplifying beamforming systems. While semi-active tags offer distinct advantages in terms of range and reliability due to their onboard power, they also introduce complexities and costs that could be mitigated by integrating more passive tag solutions. By shifting towards a system using passive tags, not only are the initial deployment costs significantly reduced, but the lifecycle and maintenance demands of the system are also decreased.

Semi-active tags, with their embedded batteries and intricate electronics, contribute unevenly to the environmental footprint of RFID systems. The environmental considerations of using battery-powered tags are non-trivial; batteries require proper disposal and recycling processes to mitigate their impact on the environment. Transitioning towards a beamforming system that utilizes more passive or even battery-free solutions could dramatically reduce this environmental load. Such a shift not only aligns with global sustainability goals but also enhances the appeal and applicability of RFID technology across more environmentally sensitive applications.

Moreover, the requirement for ongoing maintenance, primarily battery replacements in semi-active tags, poses significant logistical challenges, particularly in extensive and complex deployments. A simplified beamforming system that reduces or eliminates these semi-active components could offer substantial operational benefits. With fewer needs for regular maintenance, the system's reliability and uptime improve. This simplification could lead to broader adoption and greater scalability, as organizations would face fewer obstacles in expanding their RFID-enabled operations. As such, advancing toward a more passive-tag-oriented beamforming system not only simplifies

technical and logistical operations but also sets a new standard for efficiency and sustainability in RFID technology.

5.2.2 Preformance Promotion for the Beamforming System

We implement transfer beamforming for its efficiency and time-saving benefits, driven by localization relationships. Despite these advantages, challenges arise, notably when transfer beamforming yields suboptimal results. To address this, it's essential to establish a robust system capable of automatically detecting and correcting these anomalies. The proposed solution involves developing an anomaly detection algorithm specifically targeted for beamforming applications. Any data points that deviate beyond predefined control limits will trigger alerts, signaling potential beamforming issues.

Upon detecting these suboptimal beamforming results, the system should activate an adaptive beamforming algorithm. This corrective measure dynamically recalibrates beamforming parameters such as the angle of arrival, phase shifts, and power levels, adapting in real-time based on immediate system feedback. Such adjustments ensure the beamforming operation remains optimal despite the inherent complexities and variabilities of the operational environment.

5.2.3 On-Chip BONN

Integrating binary optical neural networks (BONNs) onto chips represents a significant leap forward in both technology and application potential. This miniaturization facilitates high-precision manufacturing processes that greatly reduce component misalignmenta vital issue that hinders the accuracy of BONNs. Utilizing advanced, elaborate technology to ensure precise and controllable production flows is critical. This not only enhances the reliability and efficiency of these devices but also scales

down their size, making them ideal for integration into portable consumer devices like digital cameras and smartphones. Such integration promises to revolutionize these devices by enabling complex, real-time processing capabilities that were previously unobtainable in such compact forms.

Moreover, the practical implications of embedding BONNs on chips extend beyond consumer electronics. For instance, in healthcare, miniaturized BONNs can power portable diagnostic tools, bringing sophisticated medical analyses directly to the point of care without the need for bulky, traditional optical equipment. This capability could enhance advanced diagnostic techniques, making them accessible in remote or underserved regions. Additionally, as these devices become more widespread, ongoing innovations and improvements in chip fabrication techniques will likely prompt further reductions in cost and energy consumption, broadening the scope of potential applications.

By continuing to refine the integration of BONNs onto chips, we can expect not only to overcome current technological barriers but also to open new avenues for their use across various sectors. This progression will necessitate collaborative efforts across the scientific and industrial landscapes to ensure these technologies are both effective and safe for widespread adoption.

5.2.4 Extend the BONN to more appcations

To effectively extend Binary Optical Neural Networks (BONNs) to a broader range of applications, practical strategies and specific technological adaptations are required. In consumer electronics, developers can integrate BONNs into smartphones and smart glasses by focusing on miniaturization and energy efficiency. This involves redesigning the chip architecture to fit within the limited space of mobile devices while maintaining performance. Engineers must also optimize the power consumption of BONNs to ensure they do not drastically reduce battery life. Collaboration with mobile manufac-

turers early in the design process can ensure that these neural networks are seamlessly integrated into existing product lines with designed software that leverages BONN capabilities for enhanced user experiences in AR.

In the automotive sector, to incorporate BONNs into driver-assistance and autonomous driving systems, the focus should be on environmental robustness and reliability. Automotive-grade BONNs need to withstand a wide range of operational conditions, from temperature extremes to vibrations. This may require developing new packaging technologies that protect the delicate optical components of BONNs. Additionally, integrating BONNs with existing vehicle sensor systems necessitates the development of standardized interfaces that allow for easy communication and data exchange. Automakers and BONN developers should work together to create integrated systems that can process and analyze optical data in real-time, providing critical inputs for vehicle safety systems and navigation aids. By developing these targeted engineering solutions and fostering industry collaborations, BONNs can be adapted and optimized for a wide range of practical applications beyond their current uses.

References

[1] Be02-05-b - optical beam expander. https://www.thorlabs.com/thorproduct.cfm?partnumber=BE02-05-B.

[2] Best englishalphabet. https://www.kaggle.com/code/mohneesh7/character-recognition#Exploratory-Data-Analysis.

[3] Cmos. https://www.thorlabs.com/thorproduct.cfm?partnumber=CS165CU.

[4] Cr absorb. https://pubs.rsc.org/en/content/articlehtml/2019/ra/c9ra00559e.

[5] Ebl price. https:https://lab.kni.caltech.edu/Usage_Rates.

[6] Energy consumption of AI poses environmental problems. https://www.techtarget.com/searchenterpriseai/feature/Energy-consumption-of-AI-poses-environmental-problems.

[7] Quartz transmittance. https://www.quora.com/What-is-the-optical-transmission-of-quartz-Will-it-let-ultraviolet-visible-and-infrared-light-pass-through-How-much-percent-of-sunlight-pass-through-it.

[8] Rectangular function. https://en.wikipedia.org/wiki/Rectangular_function.

[9] Slm transmittance. https://www.lasercomponents.com/us/product/pluto-2-phase-lcos-slm/.

[10] English alphabets. https://www.kaggle.com/datasets/mohneesh7/english-alphabets, 2017.

[11] OctoClock CDA-2990. https://www.ettus.com/all-products/octoclock/, 2022. Accessed: 2022-03-13.

[12] PyTorch. https://pytorch.org/, 2022. Accessed: 2022-03-13.

[13] SBX 400-4400 MHz Rx/Tx (40 MHz). https://www.ettus.com/all-products/sbx/, 2022. Accessed: 2022-03-13.

[14] USRP X310. https://www.ettus.com/all-products/x310-kit/, 2022. Accessed: 2022-03-13.

[15] Alien. Alien "G" Inlay ALN-9654 / ALN-9954. https://www.alientechnology.com/products/tags/g/, 2022. Accessed: 2022-03-13.

[16] Zhenlin An, Qiongzheng Lin, Qingrui Pan, and Lei Yang. Turbocharging deep backscatter through constructive power surges with a single rf source. In *Proc. of IEEE INFOCOM*, pages 1–10. IEEE, 2021.

[17] Zhenlin An, Qiongzheng Lin, Lei Yang, Wei Lou, and Lei Xie. Acquiring bloom filters across commercial rfids in physical layer. *IEEE/ACM Transactions on Networking*, 28(4):1804–1817, 2020.

[18] Reza Arablouei and Kutluyıl Dogancay. Linearly-constrained recursive total least-squares algorithm. *IEEE Signal Processing Letters*, 19(12):821–824, 2012.

[19] Farshid Ashtiani, Alexander J Geers, and Firooz Aflatouni. An on-chip photonic deep neural network for image classification. *Nature*, 606(7914):501–506, 2022.

[20] Mihalj Bakator and Dragica Radosav. Deep learning and medical diagnosis: A review of literature. *Multimodal Technologies and Interaction*, 2(3):47, 2018.

[21] Yoshua Bengio. Practical recommendations for gradient-based training of deep architectures. In *Neural Networks: Tricks of the Trade: Second Edition*, pages 437–478. Springer, 2012.

[22] Carlos Bocanegra, Mohammad A Khojastepour, Mustafa Y Arslan, Eugene Chai, Sampath Rangarajan, and Kaushik R Chowdhury. Rfgo: a seamless self-checkout system for apparel stores using rfid. In *Proceedings of the 26th Annual International Conference on Mobile Computing and Networking*, pages 1–14, 2020.

[23] Julian Bueno, Sheler Maktoobi, Luc Froehly, Ingo Fischer, Maxime Jacquot, Laurent Larger, and Daniel Brunner. Reinforcement learning in a large-scale photonic recurrent neural network. *Optica*, 5(6):756–760, 2018.

[24] Julie Chang, Vincent Sitzmann, Xiong Dun, Wolfgang Heidrich, and Gordon Wetzstein. Hybrid optical-electronic convolutional neural networks with optimized diffractive optics for image classification. *Scientific reports*, 8(1):1–10, 2018.

[25] Feiyang Chen, Nan Chen, Hanyang Mao, and Hanlin Hu. Assessing four neural networks on handwritten digit recognition dataset (mnist). *arXiv preprint arXiv:1811.08278*, 2018.

[26] Hang Chen, Jianan Feng, Minwei Jiang, Yiqun Wang, Jie Lin, Jiubin Tan, and Peng Jin. Diffractive deep neural networks at visible wavelengths. *Engineering*, 7(10):1483–1491, 2021.

[27] Shaoyuan Chen, Shan Zhong, Siyi Yang, and Xiaodong Wang. A multiantenna rfid reader with blind adaptive beamforming. *IEEE Internet of Things Journal*, 3(6):986–996, 2016.

[28] Tingjun Chen, Mahmood Baraani Dastjerdi, Harish Krishnaswamy, and Gil Zussman. Wideband full-duplex phased array with joint transmit and receive

beamforming: Optimization and rate gains. *IEEE/ACM Transactions on Networking*, 29(4):1591–1604, 2021.

[29] Yansong Chen. 4f-type optical system for matrix multiplication. *Optical Engineering*, 32(1):77–79, 1993.

[30] Yitong Chen, Tiankuang Zhou, Jiamin Wu, Hui Qiao, Xing Lin, Lu Fang, and Qionghai Dai. Photonic unsupervised learning variational autoencoder for high-throughput and low-latency image transmission. *Science Advances*, 9(7):eadf8437, 2023.

[31] Eric Y Chow, Arthur L Chlebowski, Sudipto Chakraborty, William J Chappell, and Pedro P Irazoqui. Fully wireless implantable cardiovascular pressure monitor integrated with a medical stent. *IEEE Transactions on Biomedical Engineering*, 57(6):1487–1496, 2010.

[32] Matthieu Courbariaux, Yoshua Bengio, and Jean-Pierre David. Binaryconnect: Training deep neural networks with binary weights during propagations. *Advances in neural information processing systems*, 28, 2015.

[33] Matthieu Courbariaux, Itay Hubara, Daniel Soudry, Ran El-Yaniv, and Yoshua Bengio. Binarized neural networks: Training deep neural networks with weights and activations constrained to+ 1 or-1. *arXiv preprint arXiv:1602.02830*, 2016.

[34] Robert H Dennard, Fritz H Gaensslen, Hwa-Nien Yu, V Leo Rideout, Ernest Bassous, and Andre R LeBlanc. Design of ion-implanted mosfet's with very small physical dimensions. *IEEE Journal of solid-state circuits*, 9(5):256–268, 1974.

[35] Daniel M Dobkin. *The rf in RFID: uhf RFID in practice*. Newnes, 2012.

[36] Nasim Mohammadi Estakhri and Andrea Alu. Manipulating optical reflections using engineered nanoscale metasurfaces. *Physical Review B*, 89(23):235419, 2014.

[37] Xiaoran Fan, Han Ding, Sugang Li, Michael Sanzari, Yanyong Zhang, Wade Trappe, Zhu Han, and Richard E Howard. Energy-ball: Wireless power transfer for batteryless internet of things through distributed beamforming. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 2(2):1–22, 2018.

[38] Xiaoran Fan, Longfei Shangguan, Richard Howard, Yanyong Zhang, Yao Peng, Jie Xiong, Yunfei Ma, and Xiang-Yang Li. Towards flexible wireless charging for medical implants using distributed antenna system. In *Proceedings of the 26th annual international conference on mobile computing and networking*, pages 1–15, 2020.

[39] FCC. 47 CFR 15.247 - Operation within the bands 902-928 MHz. `https://www.law.cornell.edu/cfr/text/47/15.247`, 2022. Accessed: 2022-03-13.

[40] Arthur D Fisher, Wendy L Lippincott, and John N Lee. Optical implementations of associative networks with versatile adaptive learning capabilities. *Applied Optics*, 26(23):5039–5054, 1987.

[41] Xinwei Geng, Longyue Wang, Xing Wang, Bing Qin, Ting Liu, and Zhaopeng Tu. How does selective mechanism improve self-attention networks? *arXiv preprint arXiv:2005.00979*, 2020.

[42] Graham Gobieski, Brandon Lucia, and Nathan Beckmann. Intelligence beyond the edge: Inference on intermittent embedded systems. In *Proceedings of the Twenty-Fourth International Conference on Architectural Support for Programming Languages and Operating Systems*, pages 199–213, 2019.

[43] Elena Goi and Min Gu. Laser printing of a nano-imager to perform full optical machine learning. In *The European Conference on Lasers and Electro-Optics*, page jsi_p_3. Optical Society of America, 2019.

[44] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014.

[45] Joseph W Goodman. *Introduction to Fourier optics*. Roberts and Company publishers, 2005.

[46] Joseph W Goodman and P Sutton. Introduction to fourier optics. *Quantum and Semiclassical Optics-Journal of the European Optical Society Part B*, 8(5):1095, 1996.

[47] Emil J Gumbel and Julius Lieblein. Some applications of extreme-value methods. *The American Statistician*, 8(5):14–17, 1954.

[48] Ryan Hamerly, Liane Bernstein, Alexander Sludds, Marin Soljačić, and Dirk Englund. Large-scale optical neural networks based on photoelectric multiplication. *Physical Review X*, 9(2):021032, 2019.

[49] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[50] Lu Hou, Quanming Yao, and James T Kwok. Loss-aware binarization of deep networks. *arXiv preprint arXiv:1611.01600*, 2016.

[51] Kuo-Hsien Hsia, Ming-Guang Wu, Jun-Nong Lin, Hong-Jie Zhong, and Zh-Yao Zhuang. Development of auto-stacking warehouse truck. *J. Robotics Netw. Artif. Life*, 4(4):334–337, 2018.

[52] Yung-Chang Hsiao and Shiu-Li Huang. The 5g nr beamforming grid for next-generation electronic toll collection system. In *2022 IEEE 7th International Conference on Intelligent Transportation Engineering (ICITE)*, pages 412–416. IEEE, 2022.

[53] Shuangping Huang, Haobin Wang, Yongge Liu, Xiaosong Shi, and Lianwen Jin. Obc306: A large-scale oracle bone character recognition dataset. In *2019 International Conference on Document Analysis and Recognition (ICDAR)*, pages 681–688. IEEE, 2019.

[54] Itay Hubara, Matthieu Courbariaux, Daniel Soudry, Ran El-Yaniv, and Yoshua Bengio. Binarized neural networks. *Advances in neural information processing systems*, 29, 2016.

[55] Thamer M Jamel. Performance enhancement of adaptive beamforming algorithms based on a combination method. In *2015 IEEE 12th International Multi-Conference on Systems, Signals & Devices (SSD15)*, pages 1–6. IEEE, 2015.

[56] Joseph M Kahn, Randy H Katz, and Kristofer SJ Pister. Next century challenges: mobile networking for smart dust. In *Proc. of ACM MobiCom*, pages 271–278, 1999.

[57] Takeo Kanade, Jeffrey F Cohn, and Yingli Tian. Comprehensive database for facial expression analysis. In *Proceedings fourth IEEE international conference on automatic face and gesture recognition (cat. No. PR00580)*, pages 46–53. IEEE, 2000.

[58] Nikos Kargas, Fanis Mavromatis, and Aggelos Bletsas. Fully-coherent reader with commodity sdr for gen2 fm0 and computational rfid. *IEEE Wireless Communications Letters*, 4(6):617–620, 2015.

[59] Bryce Kellogg, Aaron Parks, Shyamnath Gollakota, Joshua R Smith, and David Wetherall. Wi-fi backscatter: Internet connectivity for rf-powered devices. In *Proc.of ACM SIGCOMM*, pages 607–618, 2014.

[60] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25, 2012.

[61] Abhishek Kumar and Ebin Deni Raj. Silhouettes for human posture recognition, 2020.

[62] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[63] Yann LeCun, Corinna Cortes, and CJ Burges. Mnist handwritten digit database. *ATT Labs [Online]. Available: http://yann.lecun.com/exdb/mnist*, 2, 2010.

[64] Jingxi Li, Deniz Mengu, Yi Luo, Yair Rivenson, and Aydogan Ozcan. Class-specific differential detection in diffractive optical neural networks improves inference accuracy. *Advanced Photonics*, 1(4):046001, 2019.

[65] Kezheng Li, Juntao Li, Christopher Reardon, Christian S Schuster, Yue Wang, Graham J Triggs, Niklas Damnik, Jana Müenchenberger, Xuehua Wang, Emiliano R Martins, et al. High speed e-beam writing for large area photonic nanostructuresa choice of parameters. *Scientific reports*, 6(1):32945, 2016.

[66] Linpeng Li, Tonghui Yang, Kun Wang, Hongwei Fan, Chengyi Hou, Qinghong Zhang, Yaogang Li, Hao Yu, and Hongzhi Wang. Mechanical design of brush coating technology for the alignment of one-dimension nanomaterials. *Journal of Colloid and Interface Science*, 583:188–195, 2021.

[67] Shuang Liang, Shouyi Yin, Leibo Liu, Wayne Luk, and Shaojun Wei. Fp-bnn: Binarized neural network on fpga. *Neurocomputing*, 275:1072–1086, 2018.

[68] Xing Lin, Yair Rivenson, Nezih T Yardimci, Muhammed Veli, Yi Luo, Mona Jarrahi, and Aydogan Ozcan. All-optical machine learning using diffractive deep neural networks. *Science*, 361(6406):1004–1008, 2018.

[69] Jun Liu, Weijian Liu, Hongwei Liu, Bo Chen, Xiang-Gen Xia, and Fengzhou Dai. Average sinr calculation of a persymmetric sample matrix inversion beamformer. *IEEE Transactions on Signal Processing*, 64(8):2135–2145, 2015.

[70] Zechun Liu, Baoyuan Wu, Wenhan Luo, Xin Yang, Wei Liu, and Kwang-Ting Cheng. Bi-real net: Enhancing the performance of 1-bit cnns with improved representational capability and advanced training algorithm. In *Proceedings of the European conference on computer vision (ECCV)*, pages 722–737, 2018.

[71] CH Loo, AZ Elsherbeni, F Yang, and D Kajfez. Experimental and simulation investigation of rfid blind spots. *Journal of Electromagnetic Waves and Applications*, 23(5-6):747–760, 2009.

[72] Patrick Lucey, Jeffrey F Cohn, Takeo Kanade, Jason Saragih, Zara Ambadar, and Iain Matthews. The extended cohn-kanade dataset (ck+): A complete dataset for action unit and emotion-specified expression. In *2010 ieee computer society conference on computer vision and pattern recognition-workshops*, pages 94–101. IEEE, 2010.

[73] Xuhao Luo, Yueqiang Hu, Xiangnian Ou, Xin Li, Jiajie Lai, Na Liu, Xinbin Cheng, Anlian Pan, and Huigao Duan. Metasurface-enabled on-chip multiplexed diffractive neural networks in the visible. *Light: Science & Applications*, 11(1):1–11, 2022.

[74] Yunfei Ma, Zhihong Luo, Christoph Steiger, Giovanni Traverso, and Fadel Adib. Enabling deep-tissue networking for miniature medical devices. In *Proceedings of the 2018 Conference of the ACM Special Interest Group on Data Communication*, pages 417–431, 2018.

[75] John Margaronis, Minas Christou, Ergina Kavallieratou, and Theodoros Tzouramanis. Gcdb: a character database system. In *Proceedings of the International Workshop on Multilingual OCR*, pages 1–7, 2009.

[76] Henry Markram, Eilif Muller, Srikanth Ramaswamy, Michael W Reimann, Marwan Abdellah, Carlos Aguado Sanchez, Anastasia Ailamaki, Lidia Alonso-Nanclares, Nicolas Antille, Selim Arsever, et al. Reconstruction and simulation of neocortical microcircuitry. *Cell*, 163(2):456–492, 2015.

[77] Lukas Mennel, Joanna Symonowicz, Stefan Wachter, Dmitry K Polyushkin, Aday J Molina-Mendoza, and Thomas Mueller. Ultrafast machine vision with 2d material neural network image sensors. *Nature*, 579(7797):62–66, 2020.

[78] EM Microelectronic. EM4325. `https://www.emmicroelectronic.com/product/epc-and-uhf-ics/em4325`, 2022. Accessed: 2022-03-13.

[79] Robert Miesen, Andreas Parr, Jochen Schleu, and Martin Vossiek. 360 carrier phase measurement for uhf rfid local positioning. In *2013 IEEE International Conference on RFID-Technologies and Applications (RFID-TA)*, pages 1–6. IEEE, 2013.

[80] George Mourgias-Alexandris, A Tsakyridis, N Passalis, Anastasios Tefas, K Vyrsokinos, and Nikolaos Pleros. An all-optical neuron with sigmoid activation function. *Optics express*, 27(7):9620–9630, 2019.

[81] Raghu Mudumbai, Joao Hespanha, Upamanyu Madhow, and Gwen Barriac. Scalable feedback control for distributed beamforming in sensor networks. In *Proceedings. International Symposium on Information Theory, 2005. ISIT 2005.*, pages 137–141. IEEE, 2005.

[82] Raghuraman Mudumbai, Ben Wild, Upamanyu Madhow, and Kannan Ramchandran. Distributed beamforming using 1 bit feedback: from concept to realization. In *Proc. of Allerton*, volume 8, pages 1020–1027. Citeseer, 2006.

[83] Hyeonseob Nam and Bohyung Han. Learning multi-domain convolutional neural networks for visual tracking. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4293–4302, 2016.

[84] Lionel M Ni, Yunhao Liu, Yiu Cho Lau, and Abhishek P Patil. Landmarc: Indoor location sensing using active rfid. In *Proc. of IEEE PerCom*, pages 407–415. IEEE, 2003.

[85] Mingjun Pei, Xuemeng Jia, Rong Fu, and Zhanwei Jiao. Robust beamforming and power minimization design in miso health monitoring system. In *Journal of Physics: Conference Series*, volume 1584, page 012011. IOP Publishing, 2020.

[86] Demetri Psaltis and Nabil Farhat. Optical information processing based on an associative-memory model of neural nets with thresholding and feedback. *Optics Letters*, 10(2):98–100, 1985.

[87] Automatic Differentiation In Pytorch. Pytorch, 2018.

[88] GC Qiao, SG Hu, TP Chen, LM Rong, Ning Ning, Qi Yu, and Y Liu. Stbnn: Hardware-friendly spatio-temporal binary neural network with high pattern recognition accuracy. *Neurocomputing*, 409:351–360, 2020.

[89] Haotong Qin, Zhongang Cai, Mingyuan Zhang, Yifu Ding, Haiyu Zhao, Shuai Yi, Xianglong Liu, and Hao Su. Bipointnet: Binary neural network for point clouds. *arXiv preprint arXiv:2010.05501*, 2020.

[90] Haotong Qin, Ruihao Gong, Xianglong Liu, Xiao Bai, Jingkuan Song, and Nicu Sebe. Binary neural networks: A survey. *Pattern Recognition*, 105:107281, 2020.

[91] Adnan Siraj Rakin, Li Yang, Jingtao Li, Fan Yao, Chaitali Chakrabarti, Yu Cao, Jae-sun Seo, and Deliang Fan. Ra-bnn: Constructing robust & accurate binary neural network to simultaneously defend adversarial bit-flip attack and improve accuracy. *arXiv preprint arXiv:2103.13813*, 2021.

[92] Mohammad Rastegari, Vicente Ordonez, Joseph Redmon, and Ali Farhadi. Xnor-net: Imagenet classification using binary convolutional neural networks. In *European conference on computer vision*, pages 525–542. Springer, 2016.

[93] Manishika Rawat, Matteo Pagin, Marco Giordani, Louis-Adrien Dufrene, Quentin Lampin, and Michele Zorzi. Optimizing energy efficiency of 5g redcap beam management for smart agriculture applications. *arXiv preprint arXiv:2404.15857*, 2024.

[94] Manuel Reza, Giovanni Serafino, Tobias Otto, Ahmad Mohammad, Hakimeh Mohammadhosseini, Leili Shiramin, Francesco Floris, Matthias Kolb, Dave Bail, Simone Gabrielli, et al. Design and performance estimation of a photonic integrated beamforming receiver for scan-on-receive synthetic aperture radar. *Journal of Lightwave Technology*, 39(24):7588–7599, 2021.

[95] Frank Rosenblatt. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review*, 65(6):386, 1958.

[96] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning representations by back-propagating errors. *nature*, 323(6088):533–536, 1986.

[97] Albert Ryou, James Whitehead, Maksym Zhelyeznyakov, Paul Anderson, Cem Keskin, Michal Bajcsy, and Arka Majumdar. Free-space optical neural network based on thermal atomic nonlinearity. *Photonics Research*, 9(4):B128–B134, 2021.

[98] Sarbagya Ratna Shakya and Sudan Jha. Challenges in industrial internet of things (iiot). In *Industrial Internet of Things*, pages 19–39. CRC Press, 2022.

[99] Bhavin J Shastri, Alexander N Tait, Thomas Ferreira de Lima, Wolfram HP Pernice, Harish Bhaskaran, C David Wright, and Paul R Prucnal. Photonics for artificial intelligence and neuromorphic computing. *Nature Photonics*, 15(2):102–114, 2021.

[100] David Silver, Thomas Hubert, Julian Schrittwieser, Ioannis Antonoglou, Matthew Lai, Arthur Guez, Marc Lanctot, Laurent Sifre, Dharshan Kumaran, Thore Graepel, et al. A general reinforcement learning algorithm that masters chess, shogi, and go through self-play. *Science*, 362(6419):1140–1144, 2018.

[101] Shashi Pal Singh, Ajai Kumar, Hemant Darbari, Lenali Singh, Anshika Rastogi, and Shikha Jain. Machine translation using deep learning: An overview. In *2017 international conference on computer, communications and electronics (comptelix)*, pages 162–167. IEEE, 2017.

[102] Skyworks. SKY65162-70LF. `https://www.skyworksinc.com/en/Products/Amplifiers/SKY65162-70LF`, 2022. Accessed: 2022-03-13.

[103] Febin P Sunny, Asif Mirza, Mahdi Nikdast, and Sudeep Pasricha. Robin: A robust optical binary neural network accelerator. *ACM Transactions on Embedded Computing Systems (TECS)*, 20(5s):1–24, 2021.

[104] Mohammad H Tahersima, Keisuke Kojima, Toshiaki Koike-Akino, Devesh Jha, Bingnan Wang, Chungwei Lin, and Kieran Parsons. Deep neural network inverse design of integrated photonic power splitters. *Scientific reports*, 9(1):1–9, 2019.

[105] Vamsi Talla, Mehrdad Hessar, Bryce Kellogg, Ali Najafi, Joshua R Smith, and Shyamnath Gollakota. Lora backscatter: Enabling the vision of ubiquitous connectivity. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 1(3):1–24, 2017.

[106] Muhammad Haroon Tariq, Ioannis Chondroulis, Panagiotis Skartsilas, Nithin Babu, and Constantinos B Papadias. mmwave massive mimo channel measurements for fixed wireless and smart city applications. In *2020 IEEE 31st Annual International Symposium on Personal, Indoor and Mobile Radio Communications*, pages 1–6. IEEE, 2020.

[107] Ali Tofik and Roy Partha Pratim. Enhancing small object encoding in deep neural networks: Introducing fast&focused-net with volume-wise dot product layer. *arXiv preprint arXiv:2401.09823*, 2024.

[108] Amirsina Torfi, Rouzbeh A Shirvani, Yaser Keneshloo, Nader Tavaf, and Edward A Fox. Natural language processing advancements by deep learning: A survey. *arXiv preprint arXiv:2003.01200*, 2020.

[109] Alkiviadis Tsimpiris, Dimitrios Varsamis, and Georgios Pavlidis. Tesseract ocr evaluation on greek food menus datasets. *International Journal of Computing and Optimization*, 9(1), 2022.

[110] Zhijun Tu, Xinghao Chen, Pengju Ren, and Yunhe Wang. Adabin: Improving binary neural networks with adaptive binary sets. In *European conference on computer vision*, pages 379–395. Springer, 2022.

[111] Kristof Vandoorne, Wouter Dierckx, Benjamin Schrauwen, David Verstraeten, Roel Baets, Peter Bienstman, and Jan Van Campenhout. Toward optical signal processing using photonic reservoir computing. *Optics express*, 16(15):11182–11192, 2008.

[112] Deepak Vasisht, Swarun Kumar, Hariharan Rahul, and Dina Katabi. Eliminating channel feedback in next-generation cellular networks. In *Proceedings of the 2016 ACM SIGCOMM Conference*, pages 398–411, 2016.

[113] Deepak Vasisht, Guo Zhang, Omid Abari, Hsiao-Ming Lu, Jacob Flanz, and Dina Katabi. In-body backscatter communication and localization. In *Proc. of ACM SIGCOMM*, pages 132–146, 2018.

[114] Pramod Viswanath, David N. C. Tse, and Rajiv Laroia. Opportunistic beamforming using dumb antennas. *IEEE transactions on information theory*, 48(6):1277–1294, 2002.

[115] Dayong Wang, Aditya Khosla, Rishab Gargeya, Humayun Irshad, and Andrew H Beck. Deep learning for identifying metastatic breast cancer. *arXiv preprint arXiv:1606.05718*, 2016.

[116] Jingxian Wang, Junbo Zhang, Rajarshi Saha, Haojian Jin, and Swarun Kumar. Pushing the range limits of commercial passive {RFIDs}. In *16th USENIX Symposium on Networked Systems Design and Implementation (NSDI 19)*, pages 301–316, 2019.

[117] Jue Wang and Dina Katabi. Dude, where's my card? rfid positioning that works with multipath and non-line of sight. In *Proc. of ACM SIGCOMM*, pages 51–62, 2013.

[118] Gordon Wetzstein, Aydogan Ozcan, Sylvain Gigan, Shanhui Fan, Dirk Englund, Marin Soljačić, Cornelia Denz, David AB Miller, and Demetri Psaltis. Inference in artificial intelligence with deep optics and photonics. *Nature*, 588(7836):39–47, 2020.

[119] Timothy Woodford, Xinyu Zhang, Eugene Chai, Karthikeyan Sundaresan, and Amir Khojastepour. Spacebeam: Lidar-driven one-shot mmwave beam management. In *Proceedings of the 19th Annual International Conference on Mobile Systems, Applications, and Services*, pages 389–401, 2021.

[120] Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*, 2017.

[121] Tao Yan, Jiamin Wu, Tiankuang Zhou, Hao Xie, Feng Xu, Jingtao Fan, Lu Fang, Xing Lin, and Qionghai Dai. Fourier-space diffractive deep neural network. *Physical review letters*, 123(2):023901, 2019.

[122] Lei Yang, Qiongzheng Lin, Chunhui Duan, and Zhenlin An. Analog on-tag hashing: Towards selective reading as hash primitives in gen2 rfid systems. In *Proc. of ACM MobiCom*, pages 301–314, 2017.

[123] Chunyu Yuan and Sos S Agaian. A comprehensive review of binary neural network. *arXiv preprint arXiv:2110.06804*, 2021.

[124] Yubin Zang, Minghua Chen, Sigang Yang, and Hongwei Chen. Electro-optical neural networks based on time-stretch method. *IEEE Journal of Selected Topics in Quantum Electronics*, 26(1):1–10, 2019.

[125] Hongyi Zhang, Shan Shui, Yuwen Wu, Qiang Yang, and Yijun Cai. Posture recognition based on the improved optical diffractive neural network. In *2020 IEEE 3rd International Conference on Electronic Information and Communication Technology (ICEICT)*, pages 783–785. IEEE, 2020.

[126] Hui Zhang, Mile Gu, XD Jiang, Jayne Thompson, Hong Cai, S Paesani, R Santagati, A Laing, Y Zhang, MH Yung, et al. An optical neural chip for implementing complex-valued neural network. *Nature communications*, 12(1):457, 2021.

[127] Pengyu Zhang and Deepak Ganesan. Enabling {Bit-by-Bit} backscatter communication in severe energy harvesting environments. In *Proc. of USENIX NSDI*, pages 345–357, 2014.

[128] Zixing Zhang, Jürgen Geiger, Jouni Pohjalainen, Amr El-Desoky Mousa, Wenyu Jin, and Björn Schuller. Deep learning for environmentally robust speech recognition: An overview of recent developments. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 9(5):1–28, 2018.

[129] Renjie Zhao, Purui Wang, Yunfei Ma, Pengyu Zhang, Hongqiang Harry Liu, Xianshang Lin, Xinyu Zhang, Chenren Xu, and Ming Zhang. Nfc+ breaking nfc

networking limits through resonance engineering. In *Proc. of ACM SIGCOMM*, pages 694–707, 2020.

[130] Tiankuang Zhou, Lu Fang, Tao Yan, Jiamin Wu, Yipeng Li, Jingtao Fan, Huaqiang Wu, Xing Lin, and Qionghai Dai. In situ optical backpropagation training of diffractive optical neural networks. *Photonics Research*, 8(6):940–953, 2020.

[131] Ying Zuo, Bohan Li, Yujun Zhao, Yue Jiang, You-Chiuan Chen, Peng Chen, Gyu-Boong Jo, Junwei Liu, and Shengwang Du. All-optical neural network with nonlinear activation functions. *Optica*, 6(9):1132–1137, 2019.